hp industry standard servers

december 2002

integration note TC021201IN

Novell NetWare 6 performance tuning guidelines for ProLiant servers

table of contents

abstract

IT (information technology) professionals and administrators are constantly under intense pressure to cut cost, protect current investments, and meet immediate performance demands without service interruption. Achieving these requirements within tight budgetary constraints is not an easy task. In order to optimize your investment, it is important the server operates at the highest performance level possible.

To achieve optimum performance, a server must contain key subsystem components in addition to being configured and tuned correctly. This document provides information on the important subsystem components to consider and suggestions on how to configure and tune them.

Given that the operating system and hardware components contribute to overall server performance, the focus of this document will be on the configuration and tuning of the following key components:

- network operating system (NOS) – NetWare 6
- hardware components – disk, network, memory, processor, bus interface

The NetBench and WebBench test results sections of this document also provide sample performance results to reinforce some of the recommendations and suggestions provided in this document.

introduction

Determining the correct hardware / software mix for the user, takes in depth knowledge of the environment in which the server will be deployed. This key element is vital in order to properly tune and configure the server to maximize performance and eliminate bottlenecks. Knowledge of the user environment includes the following aspects:

- user needs – what the user requires in terms of security, storage space, etc.
- user workload characterization – types of application that will be running
- user performance requirements – level of response times / throughput expectation
- budget – the amount of money allocated for the project

A thorough understanding of the user environment aids the IT professional or administrator in selecting the correct hardware and software mix that meets the requirements without exceeding the budget.

In this document the Ziff-Davis industry standard benchmark is used as the workload generator. The NetBench and WebBench test suites are used to simulate “real-world” file I/O requests and web application workloads respectively. The operating system platform chosen is Novell NetWare 6 and the hardware selected is the ProLiant ML570 Generation 2 (G2) server.

Although the server used in this document is the ProLiant ML570 G2, the knowledge gained as a result of this study can easily be applied to any HP servers deployed in similar environments.

understanding the server deployment environment

Generally speaking, a server can typically be deployed in any environment the user chooses. However, server performance varies depending on configuration, operating environment, and workload.

Typically, in a given platform, the server subsystems likely to be exercised are the processor, memory, network, and the disk. However, the extents to which any of these are exercised depend largely on the workload characteristics and the operating system environment in which the server is deployed. For example, under a database workload, the processor, memory, and disk subsystem of the server come under intense pressure; whereas in file and print service environments, the network, memory, and disk are intensely exercised. Similarly, under web applications, the network, memory, and processor come under intense pressure. Likewise, under Exchange/Messaging workload, the memory, processor, and disk are greatly stressed. And lastly, under an application server workload, the memory, processor, and disk are under intense pressure.

To achieve optimum server performance, it is important to understand the particular environment the server will operate in and the impact the server subsystem components selected might have on the server's overall performance. Having this knowledge enables you to fine-tune the server appropriately; thereby isolating and eliminating performance bottlenecks.

This document provides a general overview of typical server subsystem components and the tuning guidelines for deployment in a file I/O and web application environments. The Ziff-Davis NetBench and WebBench benchmarks running under NetWare 6 operating system are used as the basis for the analysis.

why use industry standard benchmark for performance analysis

Ideally, the best benchmark should be the same exact application the user or customer will be running on his/her platform. However, this is usually not possible in most situations. Therefore, most IT professionals use an industry standard benchmark application that best simulates their unique environment in order to predict how well a server will perform when deployed.

In general, benchmarks can be categorized into two main groups:

- trace driven
- execution driven

The trace driven benchmark focuses primarily on real-world performance using capture/playback "traces" of real-world applications. The Ziff-Davis benchmark test suites fall in this category. This benchmark is widespread in the industry and is believed to mimic real-world applications.

The execution driven benchmark is synthetic and tends to focus on certain aspects of the server subsystem (e.g., processor, memory, etc.) that does not usually correlate to real-world usage experience. The SpecCPU2000, for instance, is a good example of a synthetic benchmark.

The object of this discussion is not to preclude or endorse one category over the other but to highlight the differences so the user can make an informed decision as to which method accomplishes set goals.

As stated earlier, the Ziff-Davis test suites were chosen. The decision to use the Ziff-Davis Benchmark is because it is designed to mimic real-world user applications. Original equipment manufacturers (OEMs) and personal computer (PC) magazine use the Ziff-Davis test suites in their research publications. To obtain the latest copy, go to www.zdnet.com.

server subsystem components and selection guidelines

The IT professional or the system administrator is responsible for selecting which subsystem components provide sufficient bandwidth for optimal overall server performance.

Once the server is selected, the appropriate NOS is chosen to leverage the server hardware components. For instance, if a multiprocessor (MP) server is selected, then a corresponding network operating system and applications that are MP-aware should be used.

The following sections highlight examples of critical server subsystem components and how the selection you make can dramatically affect server performance.

disk subsystem

The disk subsystem consists of the disk and the controller. One of the functions of the disk subsystem is to keep the memory or cache filled with useful data. Insufficient memory leads to a low cache hit rate and slower user access time. When this occurs, performance decreases. It is a known fact that disk response time is significantly slower than memory response time.

The performance of the disk subsystem can therefore be affected by the components that constitute the subsystem. There are several elements that makeup the disk subsystem (i.e., SCSI (small computer system interface), disk, controller, drivers, bus interface, etc.). The following components are crucial during the selection process:

- SCSI
- disk
- controller
- SCSI technology

SCSI technology is a widely used standard for transferring data between the disk drive and the SCSI controller. Many types of SCSI technology are currently available and the one you chose can impact overall system performance. It is therefore recommended to select the type with the highest speed your disk drive and controller can support. Table 1 lists the different types of SCSI standards currently available.

SCSI specifications

table 1. SCSI standard specifications

SCSI standard	bus speed (MHz)	50-pin narrow (8-bit)	68-pin wide (16-bit)	maximum cable length
SCSI	5	5 MB / s		6 meters
SCSI-2	10	10 MB / s	20 MB / s	3 meters
SCSI ultra	20	20 MB / s	40 MB / s	1.5 meters
SCSI ultra2	40	40 MB / s	80 MB / s	12 meters (LVD)
SCSI ultra3	40		160 MB / s	12 meters (LVD)
SCSI ultra4	80		320 MB / s	12 meters (LVD)

There have been significant improvements in drive technology over the past decade. For instance, disk drive media speeds continue to increase. HP offers a wide range of drive spindle speeds in revolutions per minute (rpm). For instance, there are 10,000 and 15,000 rpm disk drives. The 15,000 rpm models are available in 18.2 GB and 36.4 GB with an average seek time of 3.8 ms, while the 10,000 rpm models are available in 18.2 GB, 36.4 GB, and 72.8 GB with an average seek time of 5.5 ms. It is therefore, advisable to choose the appropriate disk type with the highest speed your SCSI interface and controller can support.

Similarly, as shown in table 2, HP has both Smart Array and Non Array controllers; all with different performance and capacity. The non-arrays such as the Integrated Drive Electronics (IDE) controllers and the low-end array controllers are typically embedded on the system board.

table 2. Smart Array and Non Array controllers

Smart Array controllers	channels	cache (read/write)	max drive supported	SCSI protocol supported	compatibility table
Smart Array 5312	2	128	28	Ultra3 Ultra2	
Smart Array 5304	4	256 128	56	Ultra3 Ultra2 Wide Ultra	support for ADG
Smart Array 5302	2	128 64 32*	28 56**	Ultra3 Ultra2 Wide Ultra	optional support for ADG
Smart Array 4250ES	3	64	21	Ultra2 Wide Ultra	
Smart Array 532	2	32*	20	Ultra3 Ultra2	
Smart Array 431	1	16	14**	Ultra3 Ultra2 Wide Ultra	
Smart Array 5i plus	2	64*	20	Ultra3 Ultra2	
Smart Array 5i	2	32*	20	Ultra3 Ultra2	
Integrated Smart Array	1	16*	***	Ultra2 Wide Ultra	

*- used for code, transfer buffers, and non-battery backed cache

**- based on use of HP storageWorks 14 HDD enclosure

***- depends on server

For more information regarding HP Array controllers, visit: h18000.www1.hp.com/products/servers/proliantstorage/arraycontrollers/index.html.

The Redundant Array of Inexpensive Disks (RAID) controllers can be configured in any of the following supported RAID levels depending on the performance level and data protection desired.

- RAID 0 – No data protection. Simply stripes data across multiple disk drives without any data protection. This configuration provides high level of performance at low cost.
- RAID 1 – Disk mirroring. Requires twice the amount of disk drives in order to duplicate user data for data protection. This configuration provides similar level of performance as RAID 0 in read operations. The cost is, however, higher because of the duplication of disks.
- RAID 5 – Distributed data guarding. In this configuration, the parity data is distributed across all drives. This provides protection against the failure of any one drive in the array set. It provides improved performance at a minimum cost.
- RAID Advanced Data Guarding (ADG) – In this configuration, two sets of parity data is distributed across all drives. This provides the protection against the failure of any two disk drives in the array set. This provides high fault tolerance at a minimum cost of implementation.

The disclosure of the server configurations for this document can be found in appendix e.

The disk drive, SCSI channel, and disk controller all have to communicate with each other during data transmission. Realize that these devices transfer data at different rates. It is therefore recommended to balance out the I/O devices on the buses at the right speed for optimum performance benefits.

The personal computer interconnect (PCI) / PCI-eXtension (PCI-X) bus is a 32-/64-bit industry standard interface for high-speed data transfers between peripheral components. For example, the standard PCI clock rate can be 33 or 66 MHz. On the other hand PCI-X, which is an extension of the PCI, operates at up to 133 MHz. PCI-X devices are backward compatible and adjust to the bus speed of the vendor implementation. HP servers support several implementations. Table 3 provides some examples of theoretical PCI / PCI-X transfer rates.

table 3. theoretical maximum achievable PCI / PCI-X transfer rates

clock transfer rate	PCI bus type			
	PCI (32bit)	PCI (64bit)	PCI-X (32bit)	PCI-X (64bit)
33 MHz bus	132 MB / s	264 MB / s	132 MB / s	264 MB / s
66 MHz bus	264 MB / s	528 MB / s	264 MB / s	528 MB / s
100 MHz bus			400 MB / s	800 MB / s
133 MHz bus			532 MB / s	1064 MB / s

For example, the theoretical maximum achievable throughput on a 66 MHz PCI when a 32-bit bus master bursting with a 32-bit target would be:

4 bytes per data phase x 66 million data phases per second = 264 MB / second

monitoring the disk subsystem in NetWare 6

Utility tools such as Monitor, NetWare Remote Manager (NRM), and SET parameters can be used to gauge disk subsystem performance in NetWare 6. These utilities provide clues as to whether there is a performance bottleneck in the disk subsystem. The following are parameters to monitor:

- dirty cache buffers – Number of cache buffers containing updated data that has not been written to the disk. The operating system writes the data to the disk either as soon as the cache buffer is filled or when the dirty cache delay time elapses (default 3.3 seconds).
- current disk requests – Number of pending disk I/O requests that are queued for service.
- least recently used (LRU) settings – measures the length of time it takes for a cache buffer at the head of the list to make its way down to the tail, where it becomes the LRU buffer.
- Novell Storage Services (NSS) cache buffers – Check the cache buffers used by the NSS. To view the statistics shown in table 4, type the following at the console prompt:

NSS / Cachestats

table 4. buffer cache statistics

statistics	cache
num cache buffer	512
num hash buckets	524288
min OS free cache buffers	256
num cache pages allocated	233106
cache hit percentage	76
cache hit	69803
cache miss	21624
percent of buckets used	0%
max entries in a bucket	1
total entries	1286

Check the values for the *Cache hit percentage* and the *Percent of buckets used* on the displayed result.

- Cache hit percentage provides a percentage of how the cache is performing. A percentage value of > 75 is desirable.
- Percent of buckets used provides a percentage of the total cache currently in use.

tips and tricks

Use the following tips to improve the performance of your disk subsystems:

- If the number of dirty buffers remains constant and the number of current disk requests remains high, the disk subsystem might be a bottleneck. Consider installing a faster disk drive and a controller.
- If the number of dirty buffers is frequently above 50% of total cache buffers, add more memory for cache.
- Monitor the swap activities (listed in table 5) taking place. To view the swap file, type the following console command:

  swap

table 5. virtual memory swap file (sizes in millions of bytes)

volume name: SYS	
swap file	volume
size: 22	size: 4116
used: 0	free space: 2889
min: 2	min free: 5
max: 4116	
Summary	
total swap file	total volume
size: 22	size: 4116
used: 0	free space: 2889
min: 2	min free: 5
max: 4116	

- If necessary, create a swap file (one per volume) by typing the following console command:

  SWAP ADD SYS1 MIN=4, MAX=8, MIN FREE=3000

  MIN = minimum swap file size (default = 2)

  MAX = maximum swap file (default = free volume space)

  MIN FREE = minimum free space to be preserved on a volume outside the swap file (default = 5).

- To remove a swap partition, type the following console command:

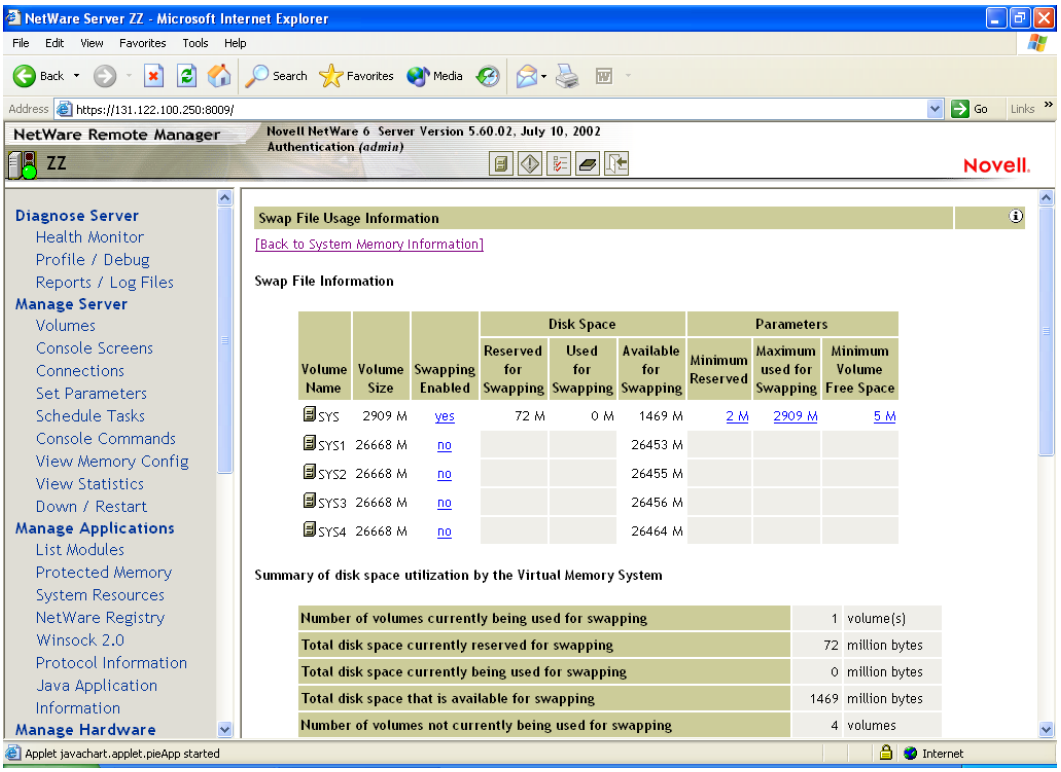
  SWAP DEL SYS1

The Swap file is a storage location for data moved to the disk by the virtual memory manager. To monitor Swap file activities using the NRM tool, follow these steps:

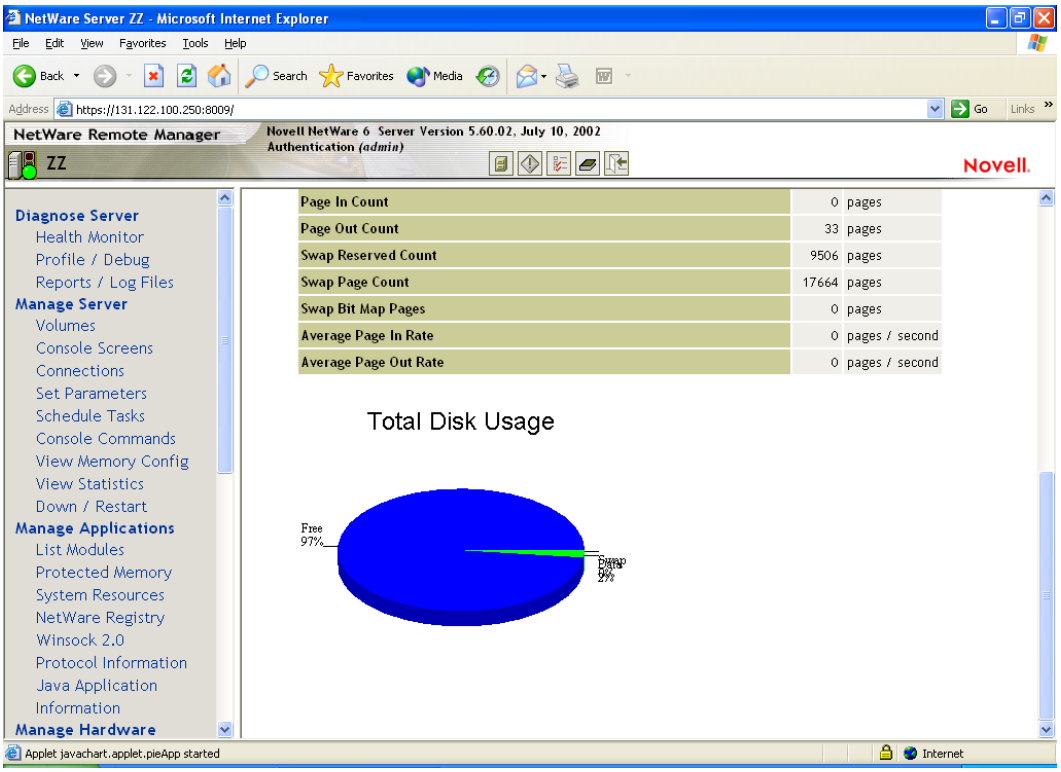
1. Open the NRM and select *View Memory Config* listed under the *Manage Server* category.

   Note: For detailed instructions on how to invoke the NRM, refer to the NetWare Remote Manager section of this document.

2. Select *Swap File Size* in the *System Memory Information* window to view the swap file usage information.



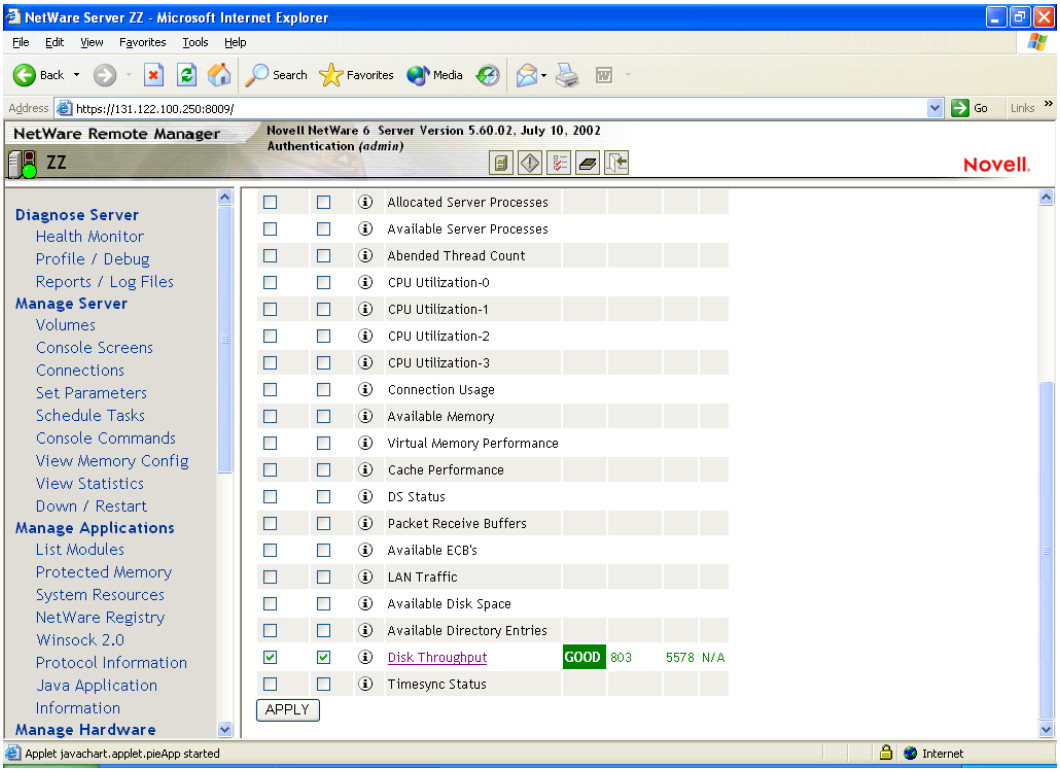
3. Scroll down with the right arrow to display the Total Disk Usage in Pie chart as shown below.



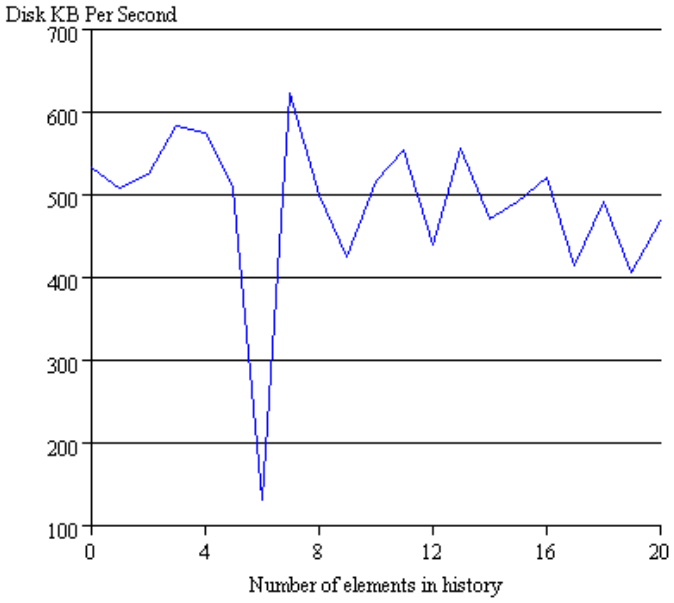
To monitor disk activities using the NRM tool, follow these steps:

1. Open the NRM, and then select *Health Monitor* listed under the *Diagnose Server* category to display the following screen.

   Note: For detailed instructions on how to invoke the NRM, refer to the NetWare Remote Manager section of this document.



2. Check *Disk Throughput* (as shown in the figure above) and click *Apply*.
3. Click on *Disk Throughput* for a graphical display of the *Disk Throughput* information as shown in the next figure.



network subsystem

The network subsystem is essentially the server's interface to other computers (i.e., LAN clients). One of its primary functions is to move data between the network and system memory as fast as possible. The network subsystem is crucial in heavy file I/O and print applications, where the majority of the request sizes are of small record sizes.

Therefore the performance of the network subsystem can be impacted by the selection and configuration of the components that makeup the subsystem. The main network subsystem components are:

- topology – Ethernet, Token Ring, Fiber Distributed Data Interface (FDDI), Asynchronous Transfer Mode (ATM), etc.
- network interface card (NIC) – 32 bit vs. 64 bit, PCI NIC vs. PCI-X NIC
- network design – switches vs. hubs

The Ethernet topology is a widely used standard topology in local area networks (LANs) compared to Token Ring, FDDI or ATM. The Ethernet technology has evolved rather rapidly over the years. It is less expensive to implement and it is backward compatible. Token Ring has become less popular due bandwidth limitation and cost. FDDI is a high speed interface, however, it is not as widely used in LANs because it is complicated and expensive to implement. The ATM, on the other hand, is mainly used as backbones and for wide area networks (WAN).

Ethernet NICs are available in a variety of types. There are PCI / PCI-X models with support for 10/100/1000 Mbps speed. The 802.3ae standard for 10-Gigabit Ethernet equipment was adopted by the Institute of Electrical and Electronics Engineers, Inc. (IEEE) in June 2002. For more information on 802.3ae, visit grouper.ieee.org/groups/802/3/ae/.

HP has many different NICs with different speeds, bandwidth, and chipsets as shown in table 6. Thus, they do have different features and varying degrees of performance and cost associated with them. For detailed information and product ordering visit: ftp.compaq.com/pub/products/servers/networking/model-compare.pdf.

table 6. hp available NICs

model number	description	transfer rate	bandwidth / bus MHz	chipset
NC3123	single port PCI 10/100 (WOL)	10, 100 Mbps	32 bit / 33 MHz	Intel 82559
NC3133	single 100FX upgrade module	100 Mbps	32 bit / 33 MHz	Intel 82558
NC3134	dual port 10/100 base adapter	10, 100 Mbps	64 bit / 66 MHz	Intel 82559
NC3135	dual port 10/100 upgrade module	10, 100 Mbps	32 bit / 33 MHz	Intel 82559
NC6132	single port 1000 SX upgrade module (fiber)	1000 Mbps	64 bit / 33 MHz	Intel 82542
NC6133	single port 1000 LX upgrade module (fiber)	1000 Mbps	64 bit / 33 MHz	Intel 82542
NC6136	single port 1000 SX (fiber)	1000 Mbps	64 bit / 33 MHz	Intel 82543GC
NC7770	PCI-X single port copper Gigabit (copper)	10, 100, 1000 Mbps	64 bit / 133 MHz	Broadcom 5701(h)
NC7131	single port 10/100/1000-T (copper)	10, 100, 1000 Mbps	64 bit / 66 MHz	Intel 82543GC
NC7132	10/100/1000-T upgrade module for NC3134 (copper)	10, 100, 1000 Mbps	64 bit / 33 MHz	Intel 82543GC

Likewise, HP has a variety of cost effective Ethernet switches with performance ranging from 10 Mbps to 10 / 100 Mbps to 10 / 100 / Gigabit that are either managed or unmanaged. Therefore, it is recommended to match-up the switches with the right interface devices for maximum performance benefits. Also, be sure to extend the network traffic across multiple buses on separate LAN segments to maximize throughputs and minimize collisions.

For more information on HP access/distribution switches, visit: www.hp.com/rnd/products/switches/.

monitoring the network subsystem in NetWare 6

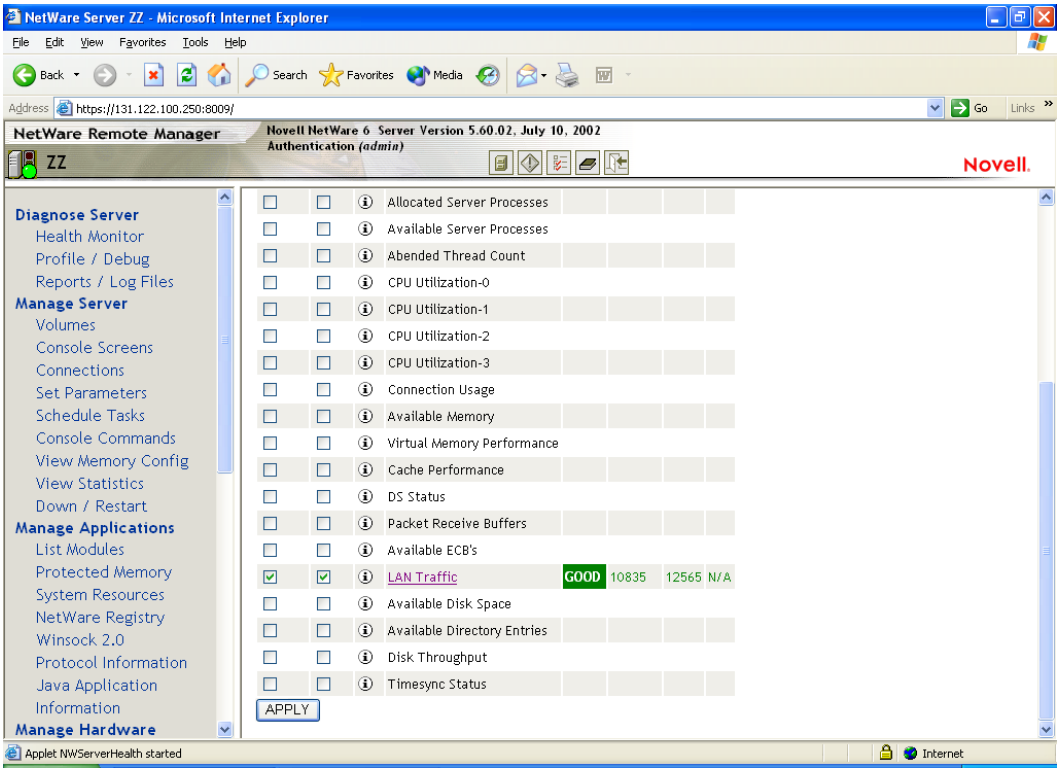
In NetWare 6, use the monitor, NetWare Remote Manager (NRM), and the SET NetWare utility tools to gauge the status of the network subsystem performance. These utilities provide clues as to whether there is a network subsystem bottleneck in your system. Some of the parameters to watch closely are:

- packet receive buffers – number of buffers that are available to the file system for holding client requests until they can be processed.
- maximum packet receive buffers – specifies the maximum number of packet buffers the operating system can allocate.
- maximum physical receive packet size –specifies the maximum packet size that can be transmitted on the network.
- NetWare Core Protocol (NCP) packet signature option – controls the NCP packet signature level on the server.

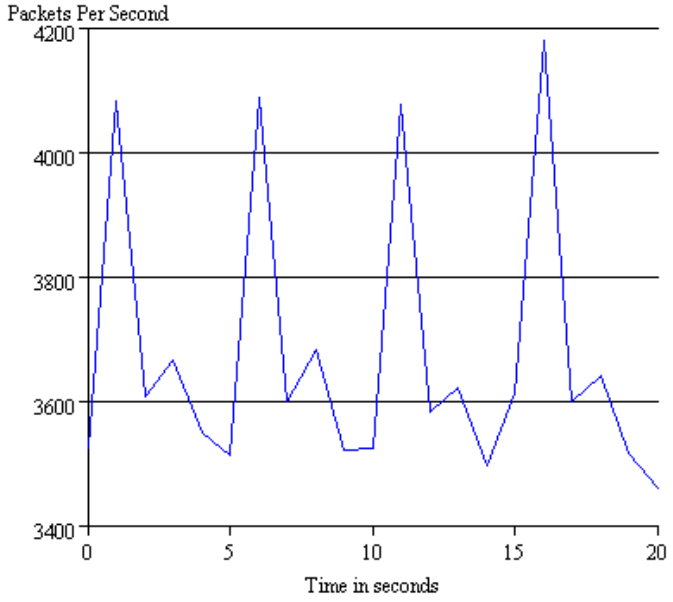
To monitor LAN Traffic using the NRM tool, follow these steps:

1. Invoke the NRM, and then select *Health Monitor* listed under the *Diagnose Server* category to display the following screen.

   Note: For detailed instructions on how to invoke the NRM, refer to the NetWare Remote Manager section of this document.



2. Check *LAN Traffic* (as shown in the figure above) and click *Apply*.
3. Click on *LAN Traffic* for a graphical display of the *Packets Per Second* as shown in the next figure.



tips and tricks

Use the following tips to improve the disk subsystem performance:

- Use fast 64-bit PCI/PCI-X Direct Memory Access (DMA) NIC devices with gigabit adapters in the server, if available.
- Increase the maximum packet receive buffers (in increments of 10), until you have one packet receive buffer per workstation.
- Set the maximum physical receive packet size to the correct size for your topology. For Token Ring and Ethernet topologies, 4202 is adequate.
- Set the NCP packet signature option to 0 to reduce network traffic congestion and server processor consumption.
- Use network monitoring device to observe network segment utilization. Add another segment, if needed.

memory subsystem

The amount of available memory for network buffers and disk I/O caching can profoundly affect the performance of a server. Typically the amount of memory in a server should be proportional to the number of users. That is, the higher the number of users, the larger the amount of memory required. Low memory, for instance, could lead to “disk thrashing” which translates to slower response times and low throughputs.

Memory is available in many different types. The differences in the technology and speed affect memory access time and the overall server performance. HP servers support several memory technologies. The goal of this section is to create an awareness of the existing types of memory technologies that are currently available. Table 7 provides information on different memory technologies and the potential performance implications.

table 7. types of memory technology

type	description
extended data out (EDO)	This memory technology was first introduced early in the year 1994 as an improvement over Fast Page Mode (FPM) memory technology. The clock rate was 40 MHz with a max bandwidth of 320 MB/s.
Synchronous Dynamic Random Access Memory (SDRAM)	This memory technology was first introduced in the early 1996 at 66 MHz. The 100 MHz version was introduced in the year1998 and in the year 1999, 133 MHz version was introduced. The unbuffered DIMMs though slightly faster than the register DIMMs has limitation in the number of DIMMs that can be interconnected on a bus due to electrical loading. The registered DIMMs on the other hand have lighter electrical load requirements and thus more DIMMs can be interconnected on a bus.
double data rate (DDR)	Introduced in the year 2000 as DDR SDRAM (100 MHz / 133 MHz). In theory DDR is capable of transferring data on both the rising and falling edges of the clock.

monitoring the memory subsystem in NetWare 6

In NetWare 6, use the NetWare Monitor and/or the all-inclusive NRM utility to check the status of the memory parameters. These utilities provide clues as to whether there is a memory bottleneck in your system. The following parameters apply to only the traditional file system. For NSS values, refer to the NSS section in this document.

- long term cache hit – cumulative percentage of the system memory since last started.
- LRU setting time – measures the length of time it takes for a cache buffer at the head of the list to make its way down to the tail, where it becomes the LRU buffer.
- SWAP activities – monitors the rate at which data is being moved from memory to disk.

tips and tricks

Use the following tips to improve the disk subsystem performance:

- Use the NRM or the NetWare monitor utility to regularly monitor total memory available and cache statistics.
- Set the long term cache hit counter range to be within 90% to 100%.
- Set the LRU setting time to greater or equal to15 minutes.
- Use the Swap utility to monitor virtual memory performance. If the overall system memory is running low, swapping does occur more often.
- Add more memory, if swapping or disk thrashing is taking place due to insufficient system memory.

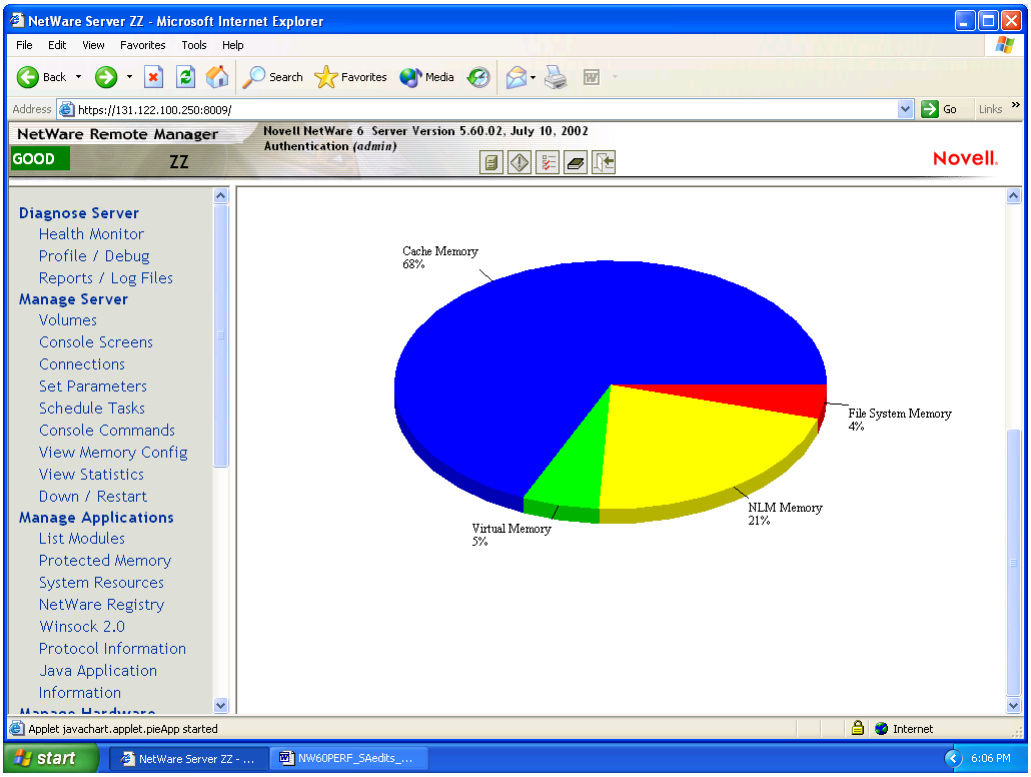
There are several tools available under NetWare for monitoring memory performance and the general health of the server. One such tool is the NetWare Monitor, which can be started by typing "monitor" on the console or through the NRM.

To monitor memory activities using the NRM tool, follow these steps:

1. Invoke the NRM and select *View Memory Config* listed under the *Manage Server* category.

   Note: For detailed instructions on how to invoke the NRM, refer to the NetWare Remote Manager section of this document.

2. Scroll down using the right arrow to display the pie chart of current memory usage (as shown in the next figure).



system processor

In Intel architecture, the processor is one of the most important hardware subsystem components. The processor is the brain of the system and thus it is involved in most of the transaction activities that occur within the server. For instance, all instructions generated by LAN attached clients must first be interpreted and processed by the processor. Thus for a server configured as a database or application server, the processor becomes critical and therefore more likely to be the source of bottleneck. Similarly in a file server environment, the processor is the heart of the file server. However, a file server is more reliant on the network, memory, and the disk subsystems for throughput capacity.

Intel processors are available in different architectural models and packages. They range from the 32-bit Pentium II, III, 4, to 64-bit Itanium 2 with front side bus (FSB) speeds of up to 533 MHz that is 128-bit wide and level 3 cache sizes of up to 3 MB. The cache can be accessed at full or half the speed of the processor in some cases. Complete details of the

In-depth information of Intel processor architecture is beyond the scope of this document.

Intel has recently introduced a relatively new technology called hyper-threading on their Xeon based family of processors. This technology exploits a program's instruction and thread-level parallelism in order to maximize overall system throughput. The ProLiant ML570 G2 server supports this technology. For more information on Intel's hyper-threading technology, visit: www.intel.com/technology/hyperthread/.

HP servers typically ship with one or more chipsets from several vendors such as Intel, ServerWorks, etc. The goal of this section is to create a performance awareness to guide the IT professional or the administrator during the decision / selection process when buying a server.

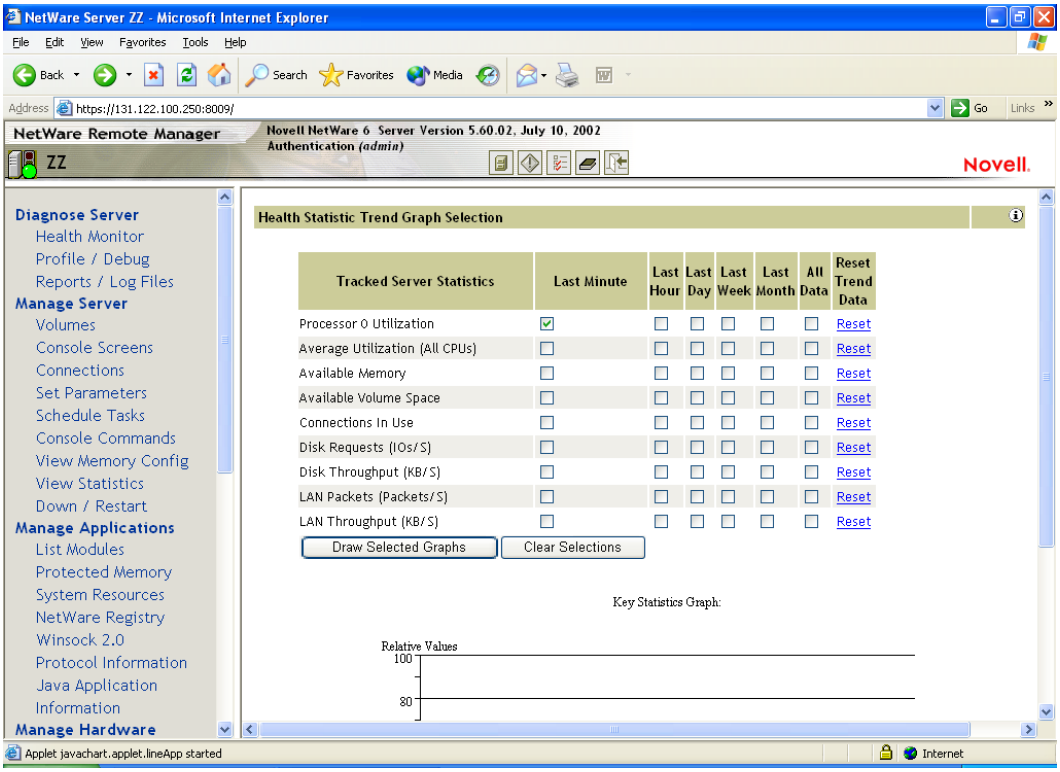
monitoring the system processor in NetWare 6

In NetWare 6, use the Monitor, NetWare Remote Manager (NRM), and the SET Parameter utility tools to gauge the status of the processor subsystem performance. These utilities provide clues as to whether there is a processor subsystem bottleneck in your system.

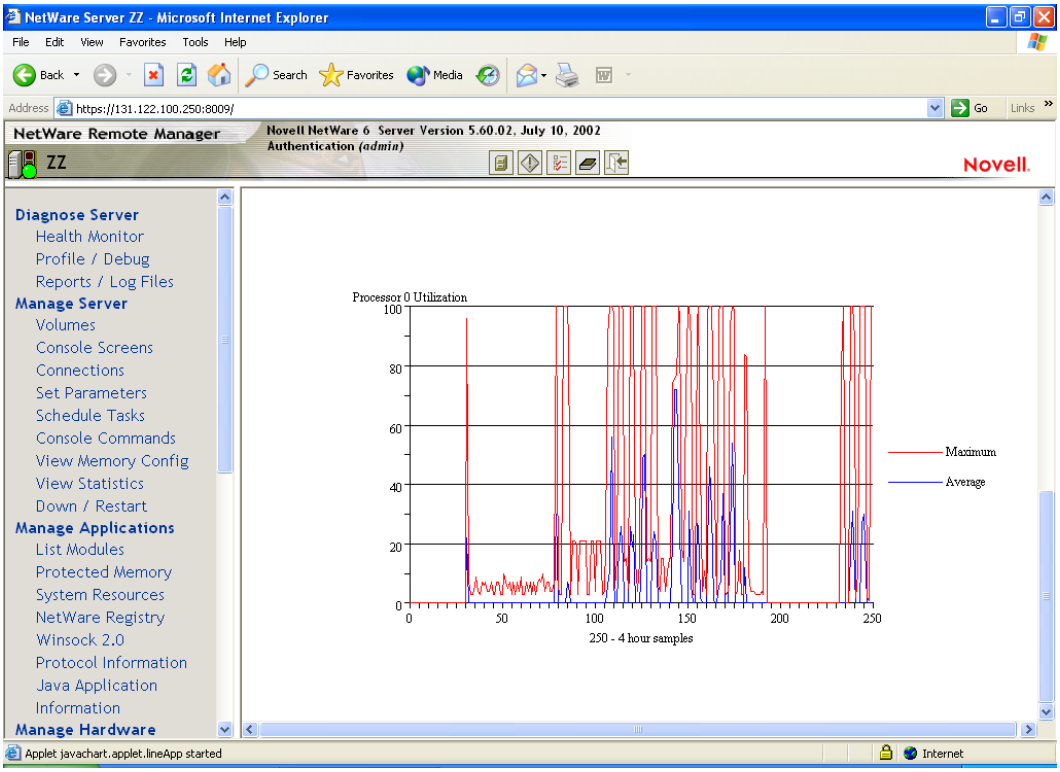
You can use the NRM tool to monitor the trend statistics of any subsystem element that is of interest. For example, to use the NRM to monitor the trend statistics of the processor, follow these steps:

1. Invoke the NRM, and then select *View Statistics* listed under the *Manage Server* category to display the following screen.

   Note: For detailed instructions on how to invoke the NRM, refer to the NetWare Remote Manager section of this document.



2. Check the time sampling interval of the trend update, and then click on *Draw Selected Graphs* for a graphical display of the selected element as shown in the next figure.



tips and tricks

Use the following tips to improve the disk subsystem performance. The processor utilization parameters to check are:

- processor utilization – a server's total processing capacity used during the last second (listed as a percentage). Check the processor utilization regularly. If the utilization is
  > 90% over a sustained period of time, it probably indicates a processor bottleneck.
- watch for errant NetWare Loadable Modules (NLMs).
- make sure the current service processes counter is adequate.
- use Direct Access Memory (DMA) devices.
- upgrade or add another processor, if the processor is the bottleneck.

There are several tools available under NetWare for monitoring processor subsystem performance and the general health of the server. Refer to the NetWare Performance Monitoring Tools section of this document for more examples.

introducing NetWare 6

According to Novell, the current version of NetWare 6 is a multithreaded, multitasking, MP-enabled operating system. A thread is a stream of control that executes instructions independently. This stream of instructions consists of tasks in the thread. A multithreaded program is where two or more threads can execute concurrently. Typically, a multithreaded program will not automatically run on more than one processor unless specified by the code developer. A multithreaded operating system with multitasking capability on the other hand can execute threads from different multithreaded programs

concurrently on a single processor by the means of a round robin technique. An MP-enabled program has the ability of executing one thread on several processors in a system at exactly the same time. In order to enhance scalability, NetWare 6 uses per-processor run queues technique for efficiency and to improve server performance. This method "affinitizes" threads to processors by allowing threads to run on the same processor they ran previously. That way, the likelihood of having useful data in the high-speed cache for immediate access by the threads is improved. For further details, refer to www.novell.com/info/collateral/docs/4621199.03/4621199.pdf.

For this document, Novell's NetWare 6 with Support Pack 2 (SP2) was used. According to Novell, NetWare 6 is optimized out-of-the-box and is also capable of tuning itself dynamically without user intervention. The results of comparing an out-of-the-box version to one with adjusted values are provided later in this document. Furthermore, an explanation of the parameters and the values that were adjusted are also presented. Examples of some useful tools to monitor and make changes to the parameter values are also provided.

setup / installation checklist

The following information is a guideline to assist you after you reached the decision that NetWare is the right operating system for your platform. This list is by no means an exhaustive one.

- know the environment in which the server is being deployed.
- know what types of applications are going to be run (profile if possible).
- know the number of users and their potential usage requirements.
- identify the disk (RAID), memory, NIC, and processor requirements.
- determine if your hardware/software is Novell "YES" certified. Refer to www.developer.novell.com for tested and approved listings.
- use the latest hardware and software (device drivers) from all vendors you plan to use. Visit their website to check for their current product offerings.
- apply the latest support pack and updates.

tuning guidelines

According to Novell, NetWare 6 is optimized out-of-the-box by default. However, depending on the application platform and/or customer needs, you can adjust many parameter values in order to optimize the performance of your server.

First collect a baseline measurement before making any adjustments to the values in the parameter lists. That way, you will be able to evaluate whether the changes you made had any effects on the performance objectives you're trying to achieve. Some suggestions are as follows:

- define your performance objectives or requirements.
- record your baseline measurements.
- use the NetWare monitoring tools to regularly observe how the server is performing. That way, you will be able to spot any imbalance before it becomes a server bottleneck.
- collect and analyze your measurement results in order to pin point a bottleneck. Once a bottleneck has been identified, fix it.
- only make one change at a time. Run your benchmark, analyze and record the results and compare it with the baseline measurement.
- repeat the above exercise until the desired performance levels or objectives are achieved.

NetWare performance monitoring tools

There are several NetWare monitoring tools that can be used either separately or in conjunction with each other for viewing server statistics, health, activities and adjusting parameters to optimize a NetWare 6 server. These tools are also an important diagnostics aid for trouble-shooting and eliminating performance bottlenecks within the server.

This section focuses on the tools used to monitor the processor, memory, disk, and the network subsystem performance and to make changes to SET parameters values to maximize server throughputs. The tools discussed in this section are as follows:

- Monitor
- NetWare Remote Manager (NRM)
- VTune

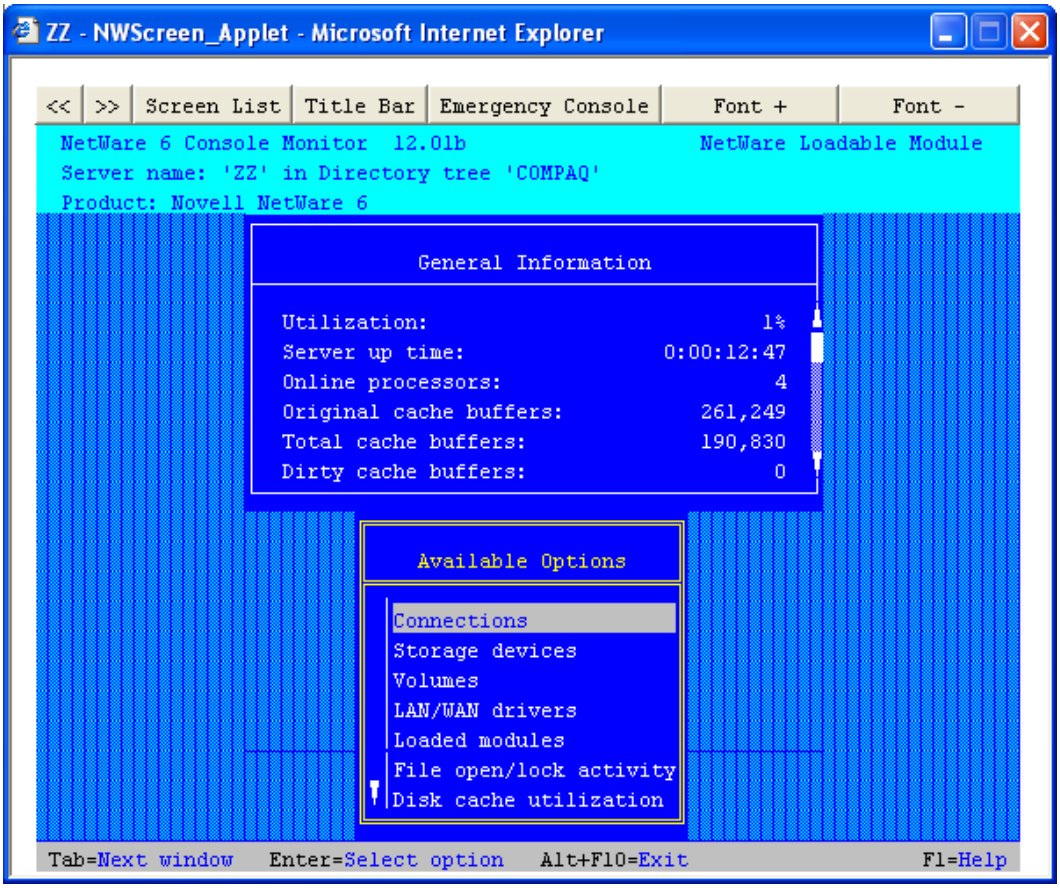
Monitor

Monitor is typically used at the server console to display status information and statistics to help manage the server, assess server memory and processor utilization. This tool also allows the user to set server parameters without returning to the console prompt to use the SET command.

To start this tool (shown in figure 1), type the following at the server console:

monitor

figure 1. NetWare Monitor tool



tips and tricks

When using the Monitor tool, use the following tips:

- toggle between the *General Information* and *Available Options* windows by using the **Tab** key.
- the arrow to the left of the vertical line in the *Available Options* window indicates the menu can be scrolled.
- press the **F1** key to access the online help and explanations of the entries listed in the *General Information* window.

performance parameters

Table 8 provides a description of each performance parameter listed on the *General Information* window. By understanding these parameters, the administrator should be able to proficiently tune the server.

table 8. General Information window performance parameters

menu option	description
utilization	The average percent processor utilization during the last second. The reminder is spent in the idle loop.
server up time	The elapse time since the server was most recently started.
online processors	Total number of active processors.
original cache buffers	The amount of cache buffers available at boot time. A cache buffer is 4 K memory page
total cache buffers	The number of cache buffers currently available for file caching. The number varies as memory is allocated / de-allocated to other NLMs or processes.
dirty cache buffers	Number of cache buffers with updated information that has not yet been written to the disk
long term cache hits	Cumulative percentage of disk block requests already in the cache.
current disk requests	Number of outstanding disk I/O request that are queued for service. If this number is consistently high, the disk subsystem might be the bottleneck.
packet receive buffers	Number of buffers available to the file system for holding client requests until they can be processed. This is determined by the network board. For example, the Ethernet NIC is 1536 bytes, FDDI card is 4096 bytes.
directory cache buffers	The number of buffers available to the file system for caching the most frequently requested directory entries.
maximum service processes	The maximum number of service processes (threads or task handlers) that the server can allocate to service client NCP requests. The server allocates the service processes as need (within the minimum and maximum parameters). Each service process takes up about 4096 bytes of memory. Once allocated, it cannot be de-allocated even when no longer needed.

table 8. General Information window performance parameters *(continued)*

menu option	description
current service processes	The number of service processes (threads or task handlers) that are currently allocated to service client NCP requests. As the number of requests from clients increase, the server creates more service processes until the maximum allocated is reached. When the maximum is reached, server performance might degrade due to insufficient number of service processes.
current connections	Number of licensed and unlicensed connections (both are considered active connections). A license for a NetWare network allows a user to attach to as many servers in the NDS tree as needed. An unlicensed connection does not use a license.
open files	Number of files currently being accessed by the server and other clients. Certain files, such as hidden files that support Novell eDirectory, are always open.

Table 9 provides a description of each performance parameter listed on the *Available options* window.

table 9. Available options window performance parameters

menu option	description
connections	Displays a list of active connections on the server. To select a connection, highlight the connection name, and then press **Enter** to show opened files by the connection. To sort active connections, press **F3**. Press **F8** for more options that can be performed on a connection.
storage devices	Displays a list of storage device objects registered in the media manager database.
volumes	Displays a list of mounted volumes. Highlight a volume to display its information. Press **Tab** to expand and activate the upper window.
LAN / WAN Drivers	Displays a list of LAN driver instances loaded on the server. Select an entry for more information on the object such as LAN driver version, logical board number, board instance number, node address, etc. Press **Tab** to view generic counter information.
loaded modules	Displays a list of all program modules loaded on the server. To sort the list of displayed modules, press **F3**. Press **F4** to recover unused memory pages from a highlighted module. Press **F8** for more options that can be performed on a module.

table 9. Available options window performance parameters *(continued)*

menu option	description
file open / lock activity	Displays a list of file system elements in a hierarchical file system. To locate a file, select the volume and the directories from the volume list. At the file level list, select the file.
disk cache utilization	Displays disk cache utilization statistics. Use this information to determine when to add more memory for cache. For more info, press **F1** for help while in this screen.
system resources	Displays server memory statistics including the list of the resource types defined by the operating system. Use the server memory statistics to assess server memory. Press **F1** for help while in this screen.
virtual memory	Displays server virtual memory information. Available options within this screen include known address spaces and swap files. The address spaces option displays a list of memory address spaces. The swap files option displays a list of disk files for temporary storing of data and programs running in memory. Press **F1** while in this screen for descriptions of the virtual memory information fields.
kernel	Displays server kernel information statistics pertaining to all threads, processors, interrupts, and busiest threads. Select any of the listed options and press **Enter** for detail option information. Press **F1** for more information about that option.
server parameters	Displays parameter categories similar to that of the SET command. Select any category listing and press **Enter** to show category fields. To edit, highlight the field and press **Enter**. For help on changing the parameter values, press **F1** while in the screen of the selected category.

For more information on the parameters listed in tables 8 and 9, visit: www.novell.com/documentation/lg/nw51/utlrfenu/data/h74jxvbx.html.

NetWare Remote Manager (Version 1.7.3)

The NRM is a versatile all-inclusive utility whose main functions can be grouped into the following activities:

- monitoring the general health of the server
- diagnostics / trouble-shooting the server
- viewing performance statistics / tuning the server

NetWare Remote Manager is used to display status information and statistics to help manage the server. This tool also allows the user to set server parameters without returning to the console prompt to use the SET command.

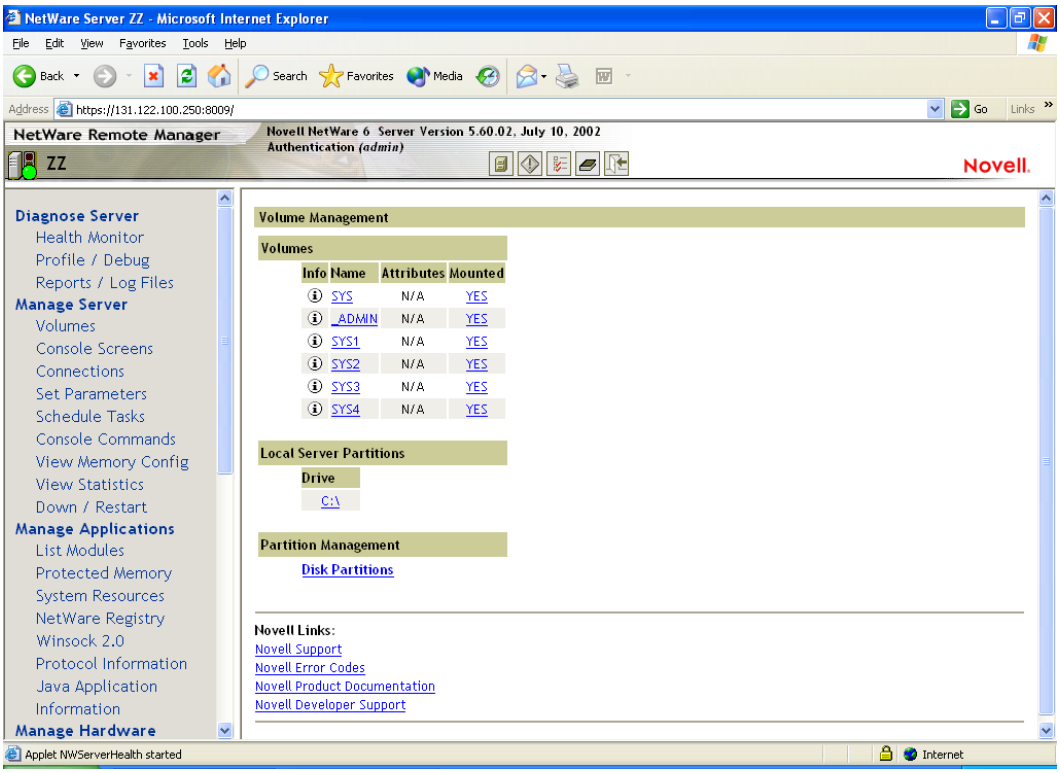
To invoked the NRM directly from the local server through *X Server Graphical Console*, follow these steps:

1. Select **Novell** from the graphical console, and then select the **Utilities** menu.
2. Click on *NetWare Remote Manager* from the drop-down menu.
3. Enter the administrator's user name and password and select **OK**.

To invoke NRM (shown in figure 2) from a remote machine, follow these steps:

1. Logon onto the server as an administrator.
2. Type in the IP address and the port number of the server at the browser and press enter to display the NRM window: https://131.122.100.250:8009/.

figure 2. NetWare Remote Manager



NRM tool utilities

Table 10 lists the NRM tool categories, a brief description of each entry, and how each one can be used.

table 10. NRM tool utilities

category	description	monitoring / viewing	diagnostics	performance / statistics
diagnose server				
health monitor	The items on this page are determined by the modules loaded on your server. Use this page to view the overall health and performance of the server. Also use this page to monitor items of interest or configure items for notification, etc.	√	√	√
profile / debug	Allows you to monitor or trouble-shoot active and suspended threads, their state, owning NLMs and execution times. Click on any thread of interest for further detailed information about the thread, such as why the thread was suspended or which nlm caused the server to abend, etc.	√	√	√

category	description	monitoring / viewing	diagnostics	performance / statistics
diagnose server (continued)				
reports / log files	Allows you to view a Server Configuration Report immediately and send the report via email. The report contains the following files: All - .ncf, .bat, .cfg, abend.log files and more.	√	√	
manage server				
volumes	Displays a pie chart of free/used information on a specific volume. It also allows you to view specific volume info such as whether it is mounted, file system name, and several other volume attributes.	√	√	
console screens	Lists all the options that can be used to access the server console screens.	√		
connections	Use this page as a connection manager to view connection info, clear connections that are not logged on and broadcast message to everyone logged on the server.	√		
set parameters	Use this page to: - view and modify set parameter values - view hidden set parameter values (Yes/No)	√		√
schedule tasks	Use this page to schedule any valid console commands (such as to run .ncf file, load nlm, send messages, etc.) you want executed on the server.	√		
console commands	Use this page to view a list of commands that can be executed on the server console.			
view memory config	Use this page to view the following system memory information: • total system memory • original cache memory • current cache memory • file system memory • reserved swap memory • swap file size • virtual memory pages • etc.	√		√

category	description	monitoring / viewing	diagnostics	performance / statistics
manage server (continued)				
view statistics	Use this page to view server statistical information on: • network management information • kernel statistical information • LSL statistical information • media manager statistical information health statistics trend graph	√		√
down / restart	Use this page to gracefully shut down, restart, and reset the server.	√		
manage applications				
list modules	Use the page to display sorted NLM information. Click on any NLM of interest for detailed information such as which flags are set, list of imported NLMs or data items.			
protected memory	Use this page to view information relating to the memory protection system on a NetWare server and a list of address spaces that currently exist on the server.	√		
system resources	Use this page to view all resource tag types in the server operating system. For detailed information on about a resource, click on the resource link.	√		√
NetWare registry	Use this page to view key registry information for the server, execute the consistency checker and flush the registry.	√	√	√
Winsock 2.0	Use this page to view Winsock 2.0 resource information, diagnostics and debug information for sockets, information for Transport Providers and Active Name Space Providers, and small set of general statistics.	√	√	√
protocol information	Use this page to view or monitor general or specific information and statistics about protocols running on the server.	√	√	√
manage hardware				
processors	Use this page to view information about each processor on the server.	√		
disk / LAN adapters	Use this page to view information about disk storage and network adapters installed on the server.	√		√

category	description	monitoring / viewing	diagnostics	performance / statistics
manage hardware (continued)				
PCI devices	Use this page to view the listing of Hardware Instance Numbers (HIN) and display the PCI configuration space for each HIN listed. There is a HIN per hardware device.	√		
other resources	Use this page to display information about drivers that have been registered with the operating system and to view their resource information.	√	√	√
manage eDirectory				
access tree walker	This page shows the NDS tree object where you can select a subordinate NDS object. Objects that contain subordinates are displayed with a '+' character preceding the object. Objects without this character are leaf objects. Selecting a leaf object shows the attribute and value information.	√		
view eDirectory partitions	This page shows a list of NDS partitions and/or replicas that exists in a tree of the server.	√		
user server groups				
build group	This page allows you to select servers you want to include into a group in order to allow you to perform similar operations on a group rather on each individual server at a time.	√		
load group file	Use this page to build a Server Group Web page from the server group configuration file specified.	√		
access other servers				
managed server list	Use this page to access any of the servers displayed on the lists. Click on the server name to access it.	√	√	√
basic file access	Used to build a list of servers in the same NDS tree. To build the list, click "Yes". If you click "Cancel" the main page of the NRM is displayed.	√	√	√

category	description	monitoring / viewing	diagnostics	performance / statistics
NetWare usage				
usage information	Use this page to display a summary report of all the unique users that have accessed the servers in the tree during the specified time interval.	√		√
configuration	Use this page for usage information configuration for all servers in the tree that are enrolled with Novell License Metering Services.	√		
advanced options	Use this page for NetWare Usage Advanced Options configuration. Click on any one of the following advanced commands to execute the following options: • recommend collector • cancel local settings • collect now • enroll now • display stats	√		

NOTE: It is important to point out that each of the elements in the categories of the table above can be drilled down further, to show detailed statistical, diagnostics, and performance metrics. To get the most value out of this utility, you are encouraged to spend good deal of time using it.

Vtune

This tool has two main components. The NetWare component (vtune.nlm) and the Intel client component (VTUNE). The vtune.nlm is loaded on the NetWare server to collect traces of information for instance, where the processor is spending its time, memory read/write misalignments or branch mis-prediction. VTUNE is then used to process the data collected by vtune.nlm. Intel's VTUNE runs under Windows 9X, Windows 2000, and Windows XP.

To collect traces using this tool, follow these steps:

1. Load the vtune.nlm from the NetWare console.

2. Select the events of interest you want to profile, and then select the sampling interval. Use the default trace file located on the root of the SYS volume or provide a name preference of your choice using the DOS 8.3 naming format and save the file.

3. Import the trace file by using the VTUNE client after you have collected all the traces needed. This allows you to view the graphical display of trace data.

You can download the 30-day evaluation version of the Intel client VTUNE at developer.intel.com/software/products/global/eval.htm

The NetWare component (vtune.nlm) can be downloaded from Novell at developer.novell.com/support/sample/tids/topt2/topt2.htm.

introducing NetBench

The performance results presented in this document are based on the current version of NetBench. NetBench is a licensed Ziff-Davis media benchmark program that measures the performance of file servers as they handle network file I/O requests from LAN attached clients. It uses LAN attached clients to generate repeated file I/O requests to a server. There are about 18 application program interface (API) routines in NetBench executed by each client in the test mix. Each client tracks the number of times the calls are executed and how long it takes to execute each call. As a result of these I/O request activities, the throughput scores and the average response time performance data are recorded. The results are then used to gauge how well the server can handle the file I/O requests generated by the clients. For more information about NetBench and on how to obtain a free copy of NetBench, visit: www.netbench.com/benchmarks/netbench/home.asp?visitor=X.

The 18 API routines NetBench executes are as follows:

- open file
- read
- write
- lock
- unlock
- get file attributes
- set file attributes
- get disk free space
- close
- get file time
- set file time
- find open
- find next
- find close
- rename file
- delete file
- create new file
- flush file buffers

In real-world, the I/O pattern and characteristics for a typical file I/O-type application is different from that of a client/server or web-type application environment. In a file and print environment for example, all requests are generated and sent from the clients to the file server for processing. The main goal of the server is to move data between the clients and the server as quickly as possible. During this process, the subsystems that endure intense pressure are:

- network
- memory
- disk

NetWare 6 has several tools and utilities for monitoring the performance of the above mentioned subsystems. The major ones include:

- NetWare Monitor
- NetWare Remote Manager
- Vtune

It is therefore prudent to regularly monitor the subsystems in order to identify potential bottlenecks in your operating environment. Typically, the processor is not the main performance bottleneck in a file and print environment, but regular monitoring should not be ignored.

Parameters that had positive effects on the overall throughput performance of the NetBench test under NetWare 6 are described in table 11.

table 11. NetBench test parameters

NSS Value	Description
CacheBalance	Set what percentage of free memory NSS will use for its buffer cache [Value=85, Range = 1 – 99]. To view the default values, type **NSS /CacheStats**
NoSalvage	Disable salvage of deleted files on the given volume.

introducing WebBench

The current version of this benchmark is WebBench 4.1. The performance results presented in this section are based on version 4.1. WebBench is a licensed Ziff-Davis Media Benchmark program that measures the performance of Web servers. It uses LAN attached clients to issue server requests for static files or a combination of static and dynamic executables. The static and/or dynamic executables are run in order to produce the data the server returns to the client. It is used to measure the performance of different Web server software packages by running WebBench against them on identical server hardware. Similarly, it is also used to measure the performance of different server hardware by running a given Web server package and WebBench against the hardware. For more information on how to obtain a free copy of WebBench, visit: www.webbench.com/benchmarks/webbench/webbench.asp?visitor=X.

Typically, the I/O workload characteristics of a website are varied based on the content of the site (static or dynamic) and the type of Web application running on the site. For example, if the site hosts mainly static content, the network, memory, and processor are subjected to intense pressure. Similarly, if the content of the site is dynamic, memory, processor, disk, and network are also affected. On the other hand, if it is e-commerce sites where compute-intensive secure socket layer (SSL) transactions are involved, then processor, memory, disk, and the network come under intense pressure.

Knowledge of these subtle but significant differences of the workload characteristics is helpful in figuring out and isolating performance bottlenecks in your environment.

NetWare 6 has several tools and utilities for monitoring the performance of the above subsystems. The major ones include:

- NetWare Monitor
- NetWare Remote Manager
- Vtune

You should regularly monitor the subsystems in order to identify potential bottlenecks in your operating environment. Depending on the types of transactions on your Web server, the processor could easily become the bottleneck and therefore should be monitored regularly as well.

NetBench test results

There is a correlation in the way a server is setup or configured and its performance. Typically, while a newer and faster server could replace a slow one, it should be the last effort in resolving the root cause of the performance issues. Poor server performance is usually a symptom of poorly configured hardware, an improperly tuned operating system, inefficient device drivers and applications, and/or slow clients.

The overall system performance can be greatly improved by using proper diagnostic tools to pinpoint bottlenecks, and then eliminate them through proper configuration and tuning. Typically, most bottlenecks in NetWare server environments are due to insufficient memory. Adding more memory in such situations usually solves the problem. However, there are other instances when considerable performance boost can only be achieved through careful planning. Planning involves reconfiguration, operating system parameter adjustments, device driver changes, and application modification to address platform specific requirements.

For this section of the document, Ziff-Davis NetBench 7.0.2 (ent_dm_nb702.tst) was used to simulate file and print I/O type requests in order to demonstrate the relative performance characteristics of the ProLiant servers.

processor scalability performance

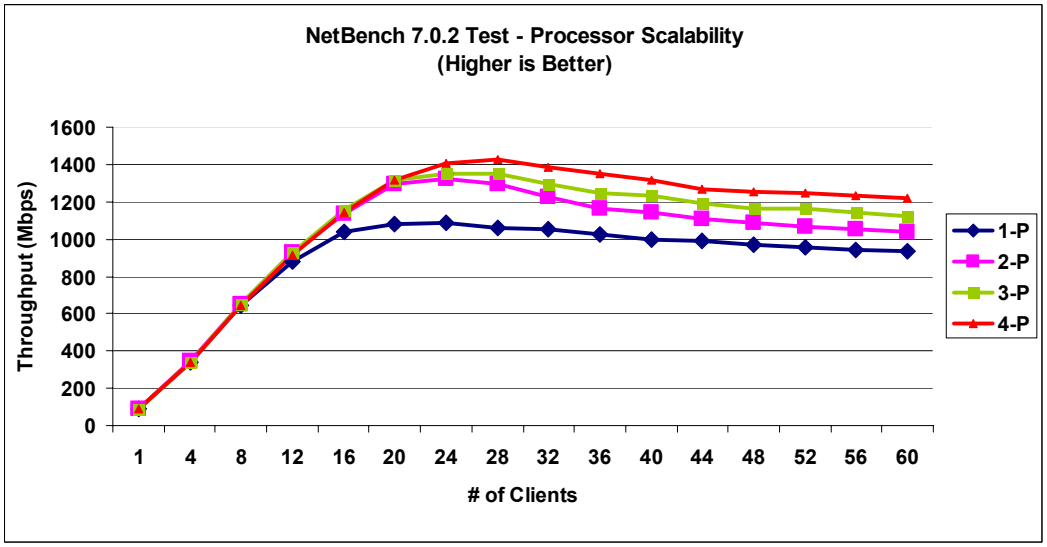
processor scalability simply means the capacity of a server to do more work at reasonable response times as you increase the number of processors.

In order to quantify the processor-scaling effects of the ProLiant ML570 G2 server, use the following commands to specify the number of processors to enable during a particular test run:

- Type **stop processors 1 2 3** on the console command prompt to enable *ONE* processor for the test.
- Type **stop processors 2 3** on the console command prompt to enable *TWO* processors for the test.
- Type **stop processors 4** on the console command prompt to enable THREE processors for the test.

Note: By default all four processors are enabled. Do not type the above commands to enable all four processors for the test.

figure 3. processor scaling (1P through 4P) on the ProLiant ML570 G2 server running NetBench



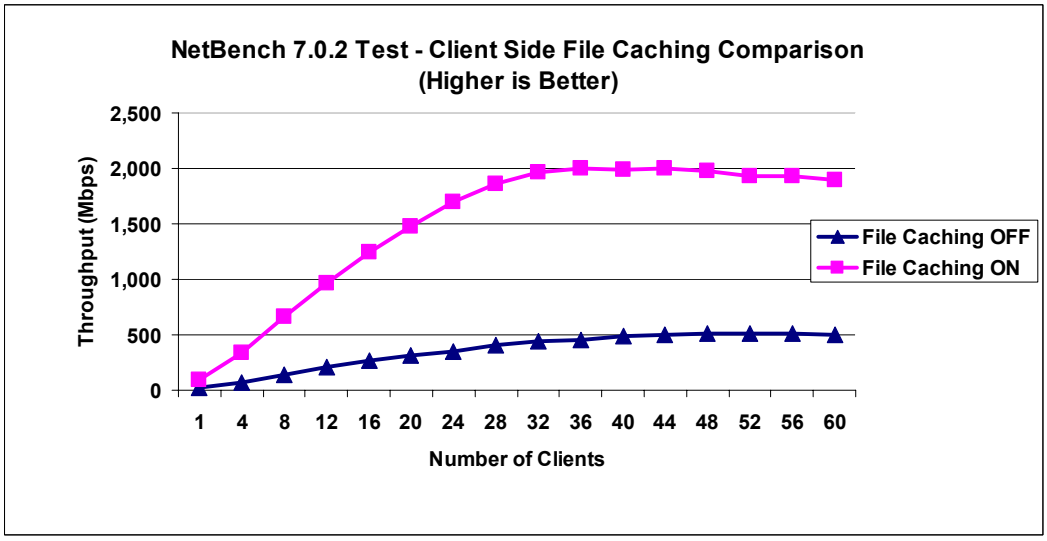
Results: Positive scaling was evident with the addition of the second, third, and fourth processors under this workload. For instance, the transaction rates did improve with the addition of the second processor by 13%. The addition of the third processor provided 5% more requests than was provided by the second processor and the fourth processor added 5% extra transactions than was provided by the third processor.

performance effects of client side file caching

In Novell Client32 for Windows, the "file caching" parameter controls whether the client will cache files locally or not. To modify the Novell Client32 for windows file caching parameter, follow these steps:

1. Click **Start**, and then select **Settings**.
2. Click on **Network and Dial-up Connections**.
3. Highlight **Local Area Connection**, and then select **Properties**.
4. Highlight **Novell Client for Windows**, and then select **Properties**.
5. Select **Advanced Settings**.
6. Under Parameter Groups, select **File Caching**.
7. Make sure it is **ON**.

figure 4. performance effects on file caching

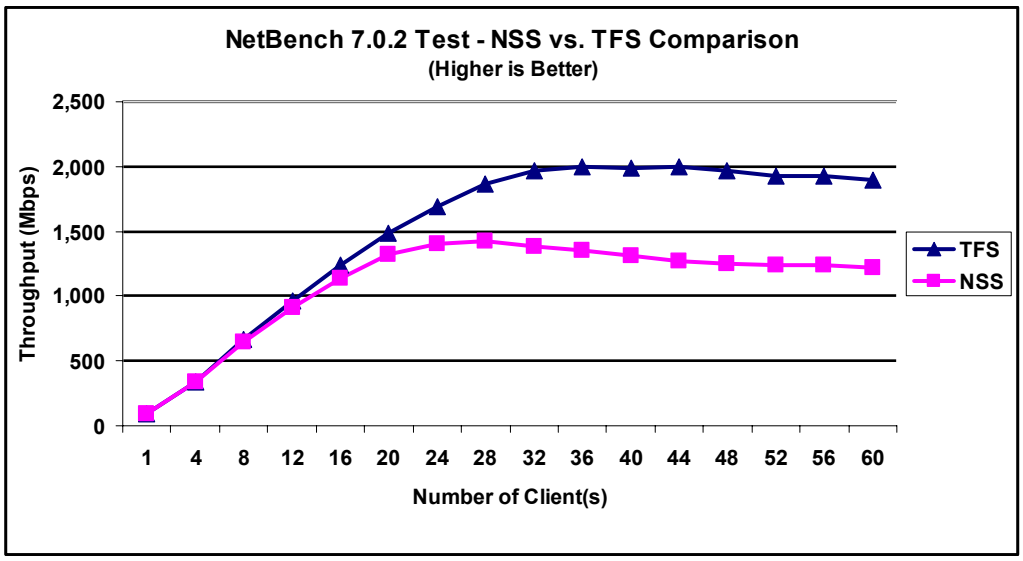


Results: Using the Novell Traditional File System (TFS) and with a workload of 60 clients, the ProLiant ML570 G2 server (shown in figure 4) provided an average 321% more throughput across the entire length of the graph with file caching turned on versus file caching turned off.

performance effects of TFS versus NSS

By default, the NSS file system is installed during the installation of NetWare 6 unless you decide to override it during normal installation. The choice of which file system to use depends on several issues, including storage requirements, security, etc. NSS is typically recommended in enterprise-type operations requiring terabytes of storage. On the other hand, the Novell Traditional File System (TFS) is Novell's original file system. It is used typically for small to medium size business operations. Figure 5 shows the performance effects of TFS compared to NSS. A more in-depth comparison of the Novell TFS and NSS is beyond the scope of this document.

figure 5. performance effects of TFS compared to NSS

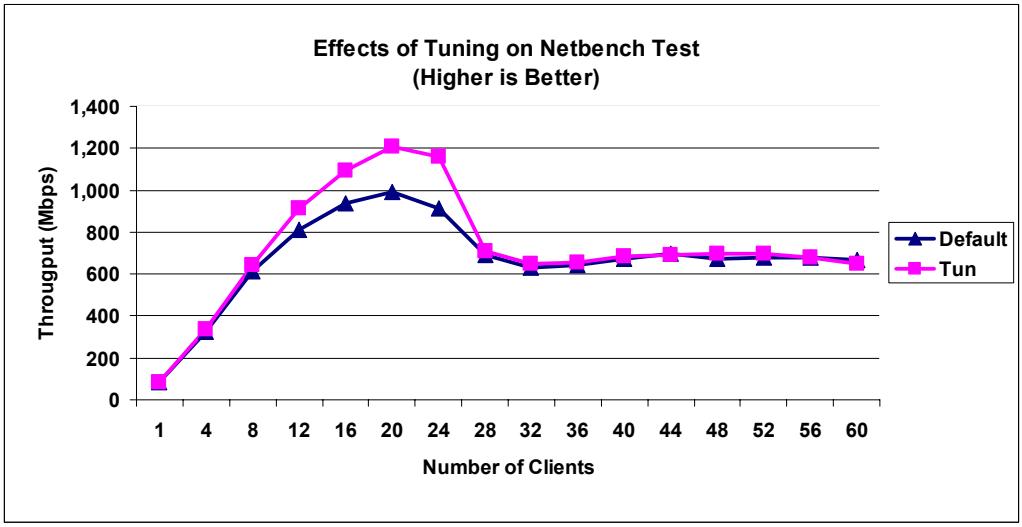


Results: With a workload of 60 clients, the ProLiant ML570 G2 server provided an average of 37% more throughput when using Novell's Traditional File System compared to NSS.

performance effects of tuning

Although NetWare 6.0 is optimized out-of-the-box for file I/O applications, minor tuning of the set parameter values was found to be beneficial to NetBench results as shown in figure 6. Refer to the autoexec.ncf and the startup.ncf for the changes made to the setup.

figure 6. performance effects of tuning



Results: With a workload of 60 clients, the ProLiant ML570 G2 server provided an average of 8% more throughput as a result of tuning compared to using the default SET parameter values.

performance effects of hyper-threading

Hyper-threading is a relatively new technology introduced by Intel on their Xeon based processor family of servers in the first quarter of 2002. The ProLiant ML570 G2 server supports this technology and can be enabled through the system BIOS. In order to leverage this technology in a multithreaded application environment, the operating system must support the Advanced Configuration and Power Interface (ACPI). The ACPI is an open industry specification co-developed by HP, Intel, Microsoft, Phoenix, and Toshiba. For more information about ACPI, visit: www.acpi.info/index.html.

When enabled, the technology presents (to the operating system) a single physical processor as two logical processors. The two logical processors, however, still share the same execution units, caches, and buses. In theory this technique can be beneficial in multithreading applications since multiple threads can be executed simultaneously with less overhead. However, preliminary results collected thus far under NetWare 6.0 (SP2) running Ziff-Davis NetBench and WebBench show unexpected degradation in performance.

HP has contacted the Novell Performance lab for further investigation. In the meantime, Novell provided the following reasons for the performance degradation when hyper-threading is enabled under NetWare 6 pending the outcome of their investigation:

1. With hyper-threading, a processor's resources (execution units, cache, etc) are shared between two logical processors. When the shared resources are busy being used by one virtual processor, they aren't available to the other virtual processor. Code that is highly optimized to use the processor efficiently doesn't see much improvement when hyper-threading is turned on because the shared resources are in use most of the time. Code that stalls the processor with complex instructions, memory accesses, and cache contention does improve when Hyper-threading is turned on because the idle shared resources are available to the other virtual shared processor. When virtual processors compete for the shared resources, their individual performance is often worse than it would have been if running uni-processor, but the total performance of the two processors is usually better than one by itself.

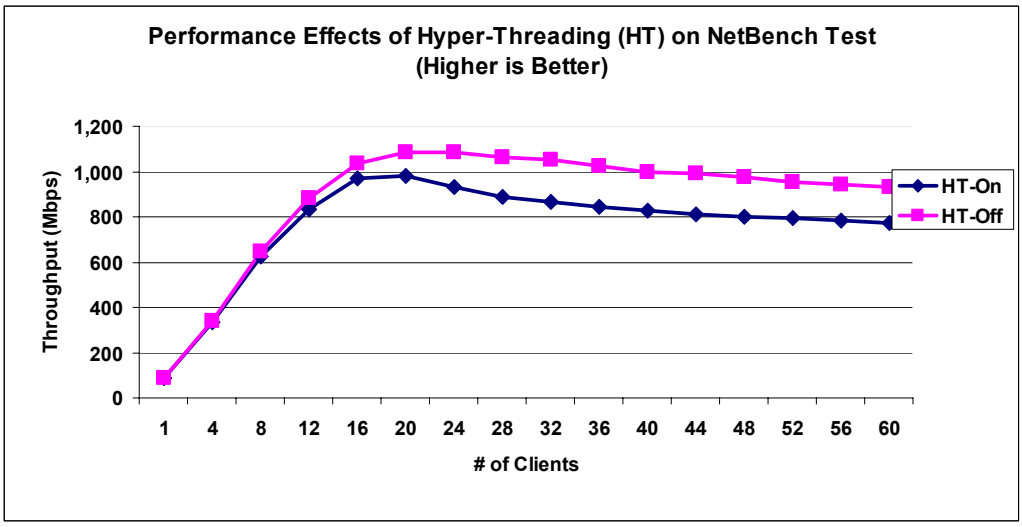
2. To improve hyper-threading performance, the OS temporarily halts processors when they are in idle; this reduces contention for the shared resources. There are two exceptions to this – the OS never halts processor zero to improve legacy Netware support, and the OS never halts processor 1 so that polling in ODI will happen regularly – it is supposed to happen even when the processor is idle. So the worst hyper-threading configuration is where processor 0 and processor 1 happen to be in the same physical package to begin with. Usually a four processor system will pair processors 0 & 4, 1 & 5, 2 & 6 and 3 & 7; this way, processor 0 and 1 don't compete with each other for the same shared resources.

3. To see a benefit with hyper-threading, you need to run applications that are:

   - Multiprocessor enabled
   - Need a lot of processor cycles
   - Often stall the processor

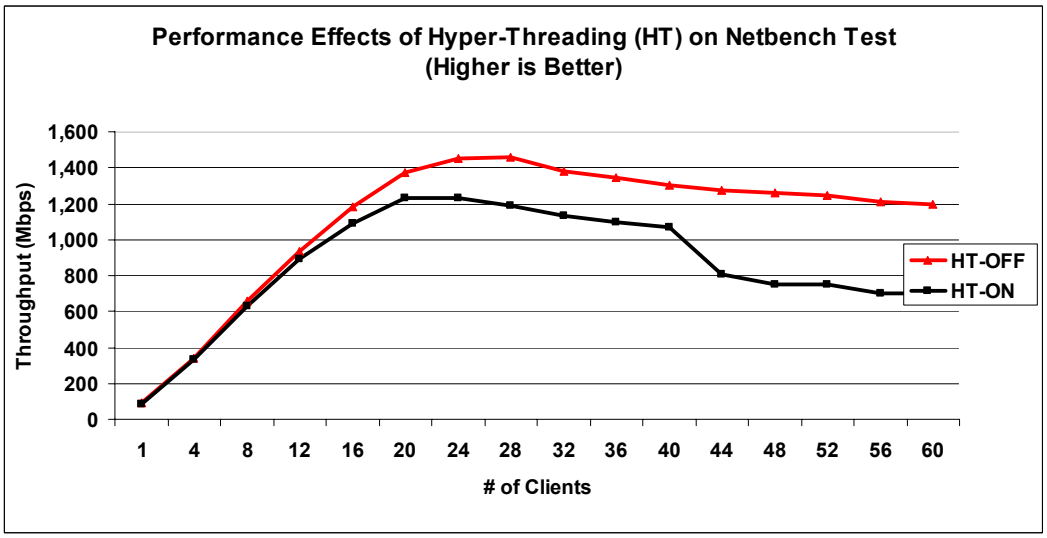
In light of the above, we are suggesting to temporarily disable hyper-threading via the system BIOS when running the Ziff-Davis benchmark tests under NetWare 6 pending the outcome of this investigation by Novell. Refer to figures 7 through 10 for the effects of hyper-threading on performance when running NetBench or WebBench with single and quad processors.

figure 7. effects of hyper-threading on NetBench performance: single processor



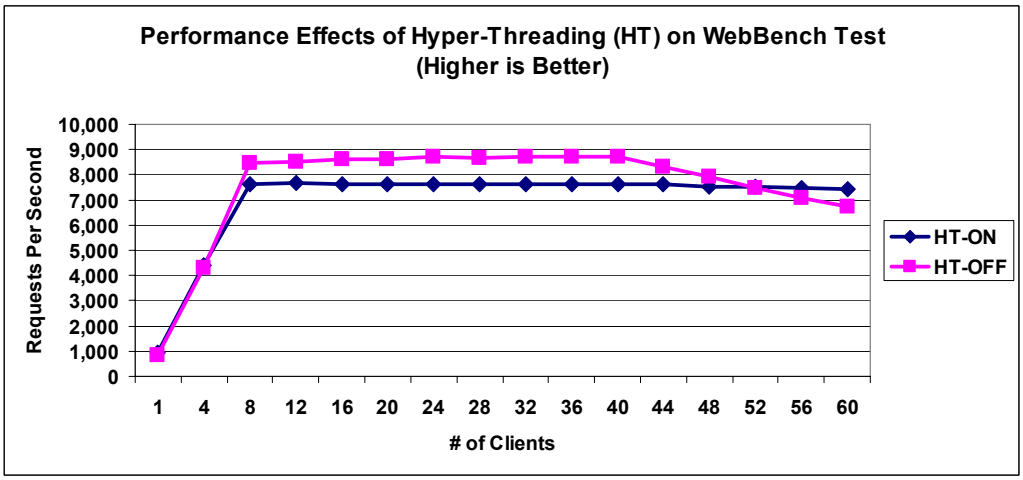
Results: With a workload of 60 clients, the ProLiant ML570 G2 server configured with a single processor provided an average of 37% more throughput with hyper-threading disabled compared to when hyper-threading was enabled.

figure 8. effects of hyper-threading on NetBench performance: quad processors



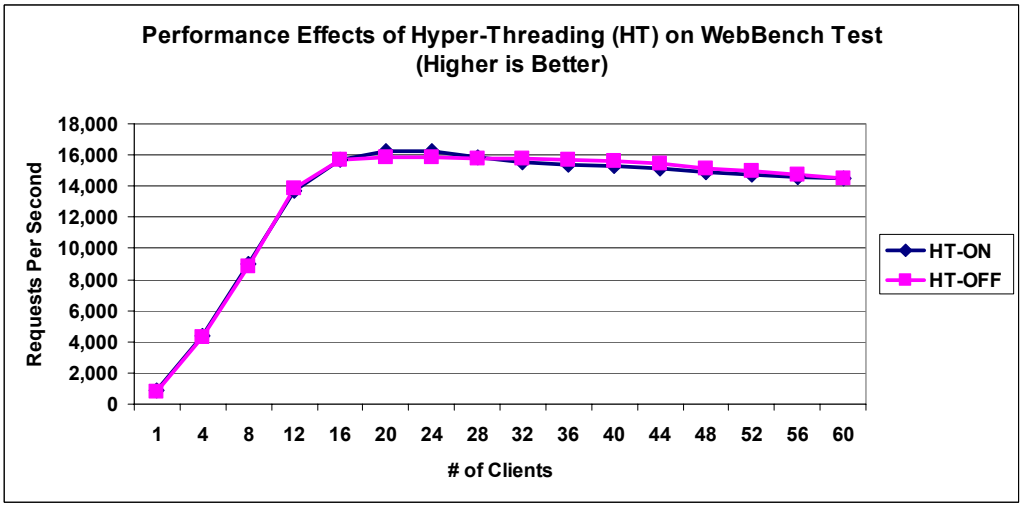
Results: With a workload of 60 clients, the ProLiant ML570 G2 server configured with four processors provided an average of 29% more throughput with hyper-threading disabled compared to when hyper-threading was enabled.

figure 9. effects of hyper-threading on WebBench performance: single processor



Results: With a workload of 60 clients, the ProLiant ML570 G2 server configured with a single processor provided an average of 8% more throughput with hyper-threading disabled compared to when hyper-threading was enabled.

figure 10. effects of hyper-threading on WebBench performance: quad processors



Results: With a workload of 60 clients, the ProLiant ML570 G2 server configured with four processors provided no statistical difference in performance on average with hyper-threading disabled compared to when hyper-threading was enabled.

Note: It should be pointed out that we have been able to document modest performance benefits when hyper-threading is enabled on 1P and 2P servers with several applications running under Microsoft Windows 2000 Server and Linux operating systems.

Table 12 lists the applications and operating systems used in the test run in figures 7 - 10.

table 12. effects of hyper-threading technology used in different operating systems

Application	W2K server	Linux 7.x & up	NetWare 6.0
SpecWeb99	Yes	Yes	No data
WebBench 4.1	Yes	Yes	No
NetBench 7.0.2	Yes	Yes	No
MS Exchange	Yes	No Data	No Data

Yes - hyper-threading is beneficial or improves server performance
No - hyper-threading is not beneficial or degrades server performance

For more information about hyper-threading technology, visit developer.intel.com/pressroom/archive/releases/20010828comp.htm.

WebBench test results

The Ziff-Davis WebBench 4.1 NetWare_simple_nsapi_wb41 was used to measure the performance of the Web server software and hardware. Each of the WebBench client issues HTTP GET requests to the server. The server responds to the requests as fast as possible by formatting the response in a readable format before sending it to the clients. The results are calibrated as transaction per second (TPS) or throughput in bytes per second.

Notes:

- The number of threads in simple_nsapi_wb41.tst script was increased from 1 thread to 5 threads in order to put more workload stress on the server.
- The workload receive buffer size was changed from the default 4 K to 32 K for the test.
- The HTTP configuration for the test was modified as follows:
  %HPPT1.0 request 0
  % persistent connection requests 100
  minimum request per PC 20
  maximum request per PC 30
  % pipeline request 0

The results of the WebBench performance tests are discussed in the following sections.

processor scalability performance

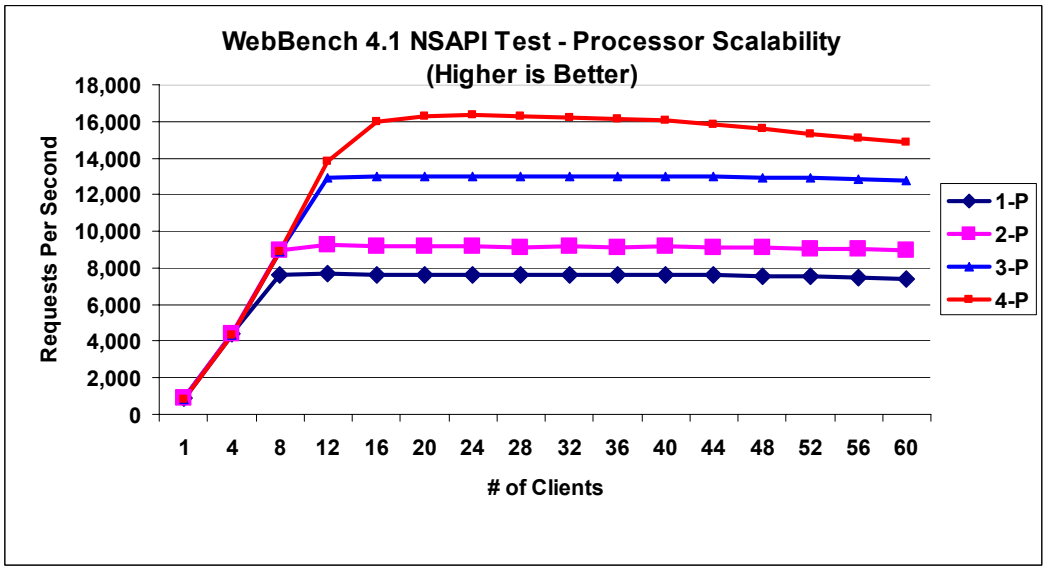
processor scalability simply means the capacity of a server to do more work at reasonable response times as you increase the number of processors.

In order to quantify the processor-scaling effects, use the following commands to specify the number of processors to enable during a particular test run:

- Type **stop processors 1 2 3** on the console command prompt to enable *ONE* processor for the test.
- Type **stop processors 2 3** on the console command prompt to enable *TWO* processors for the test.
- Type **stop processors 4** on the console command prompt to enable THREE processors for the test.

Note: By default all four processors are enabled. Do not type the above commands to enable all four processors for the test.

figure 11. processor scaling (1P through 4P) on the ProLiant ML570 G2 server running WebBench



Results: Positive scaling was evident with the addition of the second, third, and fourth processors under this workload. For instance, the transaction rates did improve with the addition of the second processor by 19%. The addition of the third processor provided 37% more requests than was provided by the second processor and the fourth processor added 19% extra transactions than was provided by the third processor.

default installation versus optimized parameters

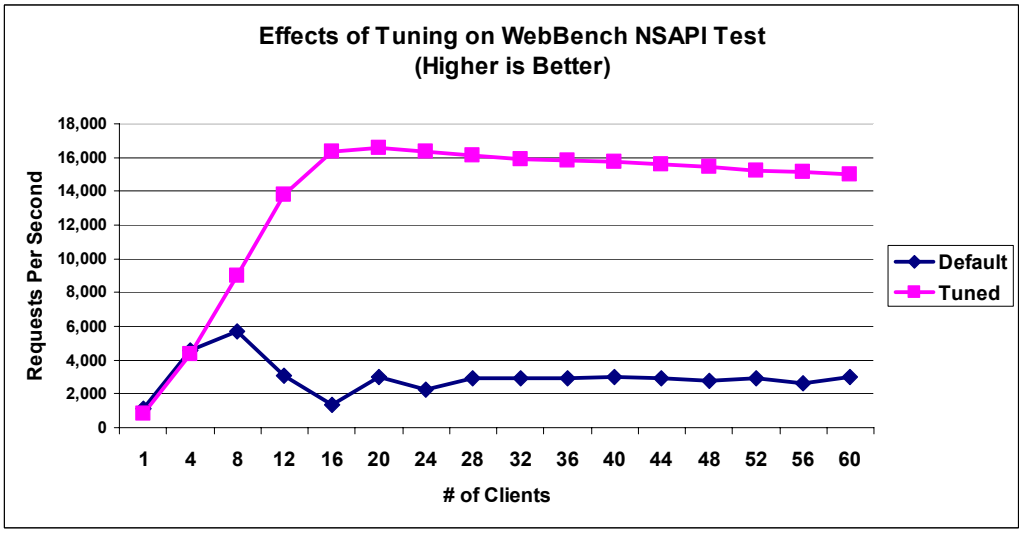
In NetWare 6, there are several parameters to be tuned in order to leverage your hardware performance capabilities. Both the NOS SET parameters and the NetWare Enterprise Web Server tuning options should be explored to in order to realize maximum performance benefits.

Refer to appendix b – NetWare 6 configuration files for details on the changes made to the following configurations files.

- Autoexec.ncf
- Startup.ncf
- Magnus.conf
- Obj.conf

Figure 12 shows the performance effects of tuning the server by selecting optimized parameters compared to accepting the default installation.

figure 12. tuning compared to default installation



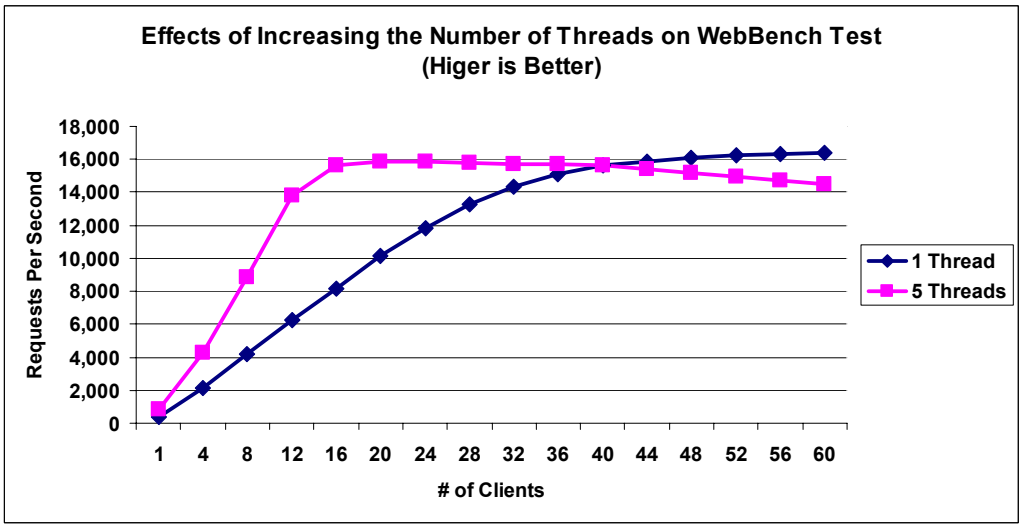
Results: With a workload of 60 clients, the ProLiant ML570 G2 server serviced on average 361% more requests when tuned compared to the results of the out-of-the-box (default) installation. The significant difference in performance is also due to the fact by default; logging is enabled whereas during tuning, it is disabled.

number of threads per client

One of the methods of increasing the workload stress on your server is by increasing the number of threads per client. That way the server can be saturated potentially faster with fewer numbers of clients. This exercise should, however, be done with caution in order to avoid negative effects on the clients due to insufficient client resources.

Figure 13 shows the performance effects of running 1 thread over 5 threads per client.

figure 13. performance effects of number of threads per client



Results: With a workload of 60 clients, the ProLiant ML570 G2 server serviced 17% more requests on average with five threads per client compared to the single thread per client test mix.

conclusion

The ProLiant ML570 G2 server is designed with the customer in mind. The rugged and modular architecture makes it the right solution to meet the customer's demands today and well into the future.

There are usually performance bottlenecks in any given server environment. The real challenge is having the necessary skills and tools to help identify and eliminate them from the system.

Obviously, the overall performance of any system can be gated by its weakest component. It is therefore, important that the server be properly tuned and balanced so that all the major subsystems are optimally configured such that one optimally configured subsystem does not impede the performance of another.

This document has highlighted the interrelationship between the subsystems (processor, memory, disk, and network) and the overall system performance in addition to the need to have a thorough understanding of the environment in which the server is going to be deployed. Tips on how to fine tune the ProLiant ML570 G2 server deployed in a file I/O intensive and Web applications environments have been included as guidelines in order to achieve optimum performance and capacity.

appendix a: server features

As shown in table 13, the ProLiant ML570 G2 server has the following key differentiators compared to a similar server configuration from other OEM vendors.

table 13. ProLiant ML570 G2 server key differentiators

Differentiator	Description
hot spare memory	Bank level failover. Up to seven banks of memory (across one or two memory banks) can be protected using this method.
hot replace memory (optional)	Available only in mirrored configuration. When two memory boards are configured as mirror during setup, a failed DIMM on a board can be replaced without bringing down the server.
mirrored memory (optional)	Two memory cards are configured as mirror of each other. All writes go to both cards while reads go to the primary memory card. All are When a failure occurs on the primary, the reads are automatically redirected to the secondary memory card.
self-healing / management features	A unique and robust management solution through a combination of self-healing features and an optional PCI based lights-out management board.

The design of the ProLiant ML570 G2 server is modular, robust, and easily serviceable so that the customer can be focused on running his business rather than the server. The system can also be upgraded and expanded with minimum cost to the customer. The intent of the rugged design is to be able to handle customer needs now and well into the future as his business expands and grows. That way, the initial investment is protected with reduction in total cost of ownership at the same time.

Table 14 highlights some key performance features of the ProLiant ML570 G2 server.

table 14. ProLiant ML570 G2 server performance features

Feature	Description
processor	1.5 GHz Galatin MP (4P max)
Chipset	ServerWorks Chipset GC HE
Memory (ship/max/type)	1 GB, 32 GB, DDR SDRAM (200 MHz)
SCSI SCSI Backplane	Adaptec 7899 Dual-Channel Ultra-3 Ultra-4
NIC	single fast Ethernet 10/100 embedded 82559 controller
PCI Slots	7 PCI-X 100 MHz 64-bits

appendix b: NetWare 6 configuration files

Table 15 includes the changes made to the autoexec.ncf and startup.ncf files while running the NetBench and WebBench tests.

table 15. NetWare 6 configuration file changes

filename	NetBench test changes	WebBench test changes
Autoexec.ncf	Enable checksum off-loading. Add "xsumrx=1 xsumtx=1 txintdelay=40 to the end of LOAD Q57.LAN line. Add "set immediate purge of deleted files = on". Add "NSS /CacheBalance = 85" (for NSS file system test). Add "NSS /Nosalvage = sys1 Add "NSS /Nosalvage = sys2 Add "NSS /Nosalvage = sys3 Add "NSS /Nosalvage = sys4 (for NSS file system test). Add "NSS /CacheBalance = 1" (for TFS test). Comment out all lines in this file that load additional programs into memory that are not needed for this test.	Enable checksum off-loading. Add "xsumrx=1 xsumtx=1 txintdelay=40 to the end of LOAD Q57.LAN line. Comment out all lines in this file that load additional programs into memory that are not needed for this test.
Startup.ncf	Comment out all IDE related entries in this file.	Comment out all IDE related entries in this file.
Magnus.conf	N/A	Add these lines: ErrLogSize 0 LoggingErrors off RqThrottle 512 RqThrottleMinPerSocket 48
Obj.conf	N/A	Add these lines: Init fn="cache-init" MaxTotalCachedFileSize="130000" MaxNumberOfOpenCachedFiles="13000" MaxNumberOfCachedFiles="13000" PollInterval="80000"

appendix c: NetBench test methodology

The performance results presented in this section of the document are based on the current version of NetBench 7.0.2. NetBench is a licensed Ziff-Davis media benchmark program that measures the performance of file servers as they handle network file I/O requests from LAN attached clients. It uses LAN attached clients to generate file I/O requests to a server. The throughput scores and the average response time performance metrics are used to gauge how well the server can handle file I/O requests from clients. For more information on NetBench, visit the website at www.netbench.com/benchmarks/netbench/netbench.asp?visitor=X.

The Enterprise version of the standard test (ent_dm_nb702) was executed on sixty clients. The local operating system (LOS) on all clients was Windows 2000 (SP2). The clients were connected to the two switches through 100 MB/s full duplex connections to two Anritsu multiflow 5048 switches. The two switches were configured as two separate LAN segments (30 clients per LAN segment). Next, the ProLiant ML570 G2 server was connected to each of the switches with a CAT5 cable to form the two LAN segments. Refer to server and test-bed tables for detailed information about the server, client, and the test-bed.

The executions of the Enterprise version of the NetBench standard test suites start with a single client issuing requests and increments the client workload by four until the number of clients issuing the requests reaches sixty.

appendix d: WebBench test methodology

The Ziff-Davis WebBench 4.1 test (NetWare_simple_nsapi_wb41) was used to measure the performance of the Web server software and hardware. For additional information on WebBench, visit www.webbench.com/benchmarks/webbench/webbench.asp?visitor=X.

The dynamic version (NetWare_simple_nsapi_wb41) was executed on sixty clients with minor modifications. For more details, refer to the notes listed later in this section of the document.

The LOS on all clients was Windows 2000 (SP2). The clients were connected to the two switches through 100 MB/s full duplex connections to two Anritsu multiflow 5048 switches. The two switches were configured as two separate LAN segments (30 clients per LAN segment). Next, the ProLiant ML570 G2 server was connected to each of the switches with a CAT5 cable to form the two LAN segments. Refer to server and test-bed tables for detailed information about the server, client, and the test-bed.

The executions of the slightly modified WebBench standard test suites start with a single client issuing requests and increments the client workload by four until the number of clients issuing the requests reaches sixty.

Notes:

- The number of threads in simple_nsapi_wb41.tst script was increased from 1 thread to 5 threads to put more workload stress on the server with the least amount of clients.
- The workload receive buffer size was changed from the default 4 K to 32 K for the test.
- The HTTP configuration for NetWare_simple_nsapi_wb41 test was modified as follows:

  %HPPT1.0 request 0

  % persistent connection requests 100

  minimum request per PC 20

  maximum request per PC 30

  % pipeline request 0

appendix e: server configuration details

Table 16 highlights the server configuration used for testing the ProLiant ML570 G2 server while running the NetBench and WebBench tests.

table 16. ProLiant server configuration

Item	Description
system BIOS and date	P32, 08/30/2002
processor	4x1.5 GHz Intel Pentium 4 Xeon (400 MHz FSB)
chipset	ServerWorks Grand Champion – High-End
L2 cache, L3 cache	256 KB, 1 MB
RAM	1024 MB (PC1600)
Smart Array controller and Firmware Version	HP Smart Array 5312-128MB, V1.86 (slot # 5)
Smart Array Controller Device Driver and Version	CPQRAID.HAM, v2.03, 05/10/02
number and type of disk drives for Smart Array controller	12 x 9.1GB Wide Ultra2 SCSI
integrated disk controller and firmware version	Compaq Ultra3 BIOS v3.02.01, (slot# 10008)
integrated disk controller device driver version	ADPT160M.HAM, v17.12.13, 06/21/02
number and type of integrated drive	1 x 4.3 GB wide ultra SCSI-3
RAID Configuration	RAID 0 (4 logical partitions)
gigabit NIC	2 x NC7770 (BC5701), slot# 6 and 7
NIC driver version and date	Q57.LAN, v2.34, 07/18/02
network operating system version and date	NetWare 6,v5.60.02, 07/10/02 (SP2)
web server version	Novell Enterprise Web Server v6.00g

appendix f: test bed details

Table 17 lists the details of the test bed used for testing the ProLiant ML570 G2 server while running the NetBench and WebBench tests.

table 17. test bed details

Item	Description
client configuration	
network switches	2 x Anritsu Multiflow 5048
number clients and configuration	60xDesktop i-Paq 500 MHz 128MB 8 GB IDE (ST38410A)
client local operating system (LOS)	Windows 2000 version 5.00.2195 (SP2)
Novell Client	Novell Client for Windows v4.83.00
client NIC	Intel 82559 Fast Ethernet
client NIC driver provider	Microsoft
client NIC driver version and Date	Version 4.1.67.0 Oct. 26, 1999
client disk driver provider	Microsoft
client disk driver version and date	Version 5.0.2183.1 Nov. 14, 1999
test type	
NetBench version	NetBench 7.0.2 (ent_dm_nb702.tst)
WebBench version	WebBench 4.1 (simple_nsapi_wb41.tst)
controller	
same as client configuration above	N/A

references

[1] Novell Technical Information Document, NetWare server setup checklist – TID2944438, March 5, 1999, pp. 1-4.

[2] Novell Technical Information Document, Performance, Tuning and Optimization Part 1 – TID10012765, May 18, 2002, pp. 1-8.

[3] Novell Technical Information Document, Performance, Tuning and Optimization – TID2943356, February 6, 2002, pp. 1-11.

[4] Novell Technical Information Document, NetWare server setup checklist – TID2944438, March 5, 1999, pp. 1-4.

[5] Intel Server Board SHG2: ISAPI ecommerce WebBench Tests, eTesting Labs, May 2002, pp. 1-8.

[6] Intel PRO/1000 Family of Gigabit Server adapters, NetWare Command Line keyword and Parameters http://www.intel.com/support/network/adapter/1000/nwkey.htm

[7] Compaq White Paper, New Adaptive RAID Level for Today's Larger Storage Capacities: Advanced Data Guarding, 13JE-1000A-WWEN, October 2000, pp. 1-4.

[8] IBM Redbook: Tuning IBM @server xserver for performance, July 2002, pp. 7-544

[9] Compaq Technical Guide: I/O Performance Tuning of Compaq Servers, March 1999, pp. 1-37.

for more information

To learn more about the Ziff-Davis test suites used in this document, visit www.zdnet.com.

To determine if your hardware/software is Novell "yes" certified, refer to www.developer.novell.com for tested and approved listings.

For more information on NetBench, visit: www.netbench.com/benchmarks/netbench/netbench.asp?visitor=X.

For more information on WebBench, visit: www.webbench.com/benchmarks/webbench/webbench.asp?visitor=X.

Hyper-threading technology information can be found at developer.intel.com/pressroom/archive/releases/20010828comp.htm.

feedback

Help us improve our technical communication. Let us know what you think about the technical information in this document. Your feedback is valuable and will help us structure future communications. Please send your comments to: OSIntegrationFeedback@hp.com.

12/2002

Document Number TC021201IN